

Документ подписан простой электронной подписью
 Информация о владельце:
 ФИО: ЧУМАЧЕНКО ТАТЬЯНА АЛЕКСАНДРОВНА
 Должность: РЕКТОР
 Дата подписания: 31.08.2022 11:48:12
 Уникальный программный ключ:
 9c9f7aaffa4840d284abe156657b8f85432bdb16



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ
ГУМАНИТАРНО-ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ЮУнГГПУ»)

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
(ОЦЕНОЧНЫЕ СРЕДСТВА)

Шифр	Наименование дисциплины (модуля)
Б1.О	Инструментальные средства информационных систем

Код направления подготовки	09.03.02
Направление подготовки	Информационные системы и технологии
Наименование (я) ОПОП (направленность / профиль)	Информационные технологии в образовании
Уровень образования	бакалавр
Форма обучения	очная

Разработчики:

Должность	Учёная степень, звание	Подпись	ФИО
Старший преподаватель			Боровская Елена Владимировна

Рабочая программа рассмотрена и одобрена (обновлена) на заседании кафедры (структурного подразделения)

Кафедра	Заведующий кафедрой	Номер протокола	Дата протокола	Подпись
Кафедра информатики, информационных технологий и методики обучения информатике	Рузаков Андрей Александрович	10	13.06.2019	
Кафедра информатики, информационных технологий и методики обучения информатике	Рузаков Андрей Александрович	1	10.09.2020	

Раздел 1. Компетенции обучающегося, формируемые в результате освоения образовательной программы с указанием этапов их формирования

Таблица 1 - Перечень компетенций, с указанием образовательных результатов в процессе освоения дисциплины (в соответствии с РПД)

Формируемые компетенции			
Индикаторы ее достижения	Планируемые образовательные результаты по дисциплине		
	знать	уметь	владеть

ОПК-2 способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности

ОПК.2.1 Знать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.	3.1 знать современные инструментальные средства для разработки информационных систем		
ОПК.2.2 Уметь выбирать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.		У.1 Уметь выбирать инструментальные средства для решения конкретных задач	
ОПК.2.3 Иметь навыки применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности.			В.1 Иметь навыки применения инструментальных средства при решении задач профессиональной деятельности.

ОПК-5 способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем

ОПК.5.1 Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем	3.2 Знать основы работы СУБД		
ОПК.5.2 Уметь выполнять параметрическую настройку информационных и автоматизированных систем.		У.2 Уметь настраивать информационные системы.	
ОПК.5.3 Иметь навыки инсталляции программного и аппаратного обеспечения информационных и автоматизированных систем			В.2 Иметь навыки инсталляции программного обеспечения информационных систем

ОПК-7 способен осуществлять выбор платформ и инструментальных программно-аппаратных средств для реализации информационных систем

ОПК.7.1 Знать основные платформы, технологии и инструментальные программно-аппаратные средства для реализации информационных систем	З.3 Знать основные платформы, технологии и инструментальные средства для реализации информационных систем		
ОПК.7.2 Уметь осуществлять выбор платформ и инструментальных средств для реализации информационных систем, применять современные технологии реализации информационных систем		У.3 Уметь осуществлять выбор инструментальных средств для разработки информационных систем	
ОПК.7.3 Иметь навыки владения технологиями и инструментальными программно-аппаратными средствами для реализации информационных систем			В.3 Иметь навыки владения инструментальными средствами для реализации информационных систем

Компетенции связаны с дисциплинами и практиками через матрицу компетенций согласно таблице 2.

Таблица 2 - Компетенции, формируемые в результате обучения

Код и наименование компетенции	
Составляющая учебного плана (дисциплины, практики, участвующие в формировании компетенции)	Вес дисциплины в формировании компетенции (100 / количество дисциплин, практик)
ОПК-2 способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	
Инструментальные средства информационных систем	14,29
Управление данными	14,29
Информационные технологии	14,29
Методы и средства проектирования информационных систем и технологий	14,29
производственная практика (технологическая (проектно-технологическая))	14,29
Большие данные	14,29
Методы искусственного интеллекта	14,29
ОПК-5 способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем	
Администрирование информационных систем	20,00
Архитектура информационных систем	20,00
Инструментальные средства информационных систем	20,00
Инфокоммуникационные системы и сети	20,00
производственная практика (технологическая (проектно-технологическая))	20,00
ОПК-7 способен осуществлять выбор платформ и инструментальных программно-аппаратных средств для реализации информационных систем	
Администрирование информационных систем	25,00
Архитектура информационных систем	25,00
Инструментальные средства информационных систем	25,00
производственная практика (технологическая (проектно-технологическая))	25,00

Таблица 3 - Этапы формирования компетенций в процессе освоения ОПОП

Код компетенции	Этап базовой подготовки	Этап расширения и углубления подготовки	Этап профессионально-практической подготовки
ОПК-2	Инструментальные средства информационных систем, Управление данными, Информационные технологии, Методы и средства проектирования информационных систем и технологий, производственная практика (технологическая (проектно-технологическая)), Большие данные, Методы искусственного интеллекта		производственная практика (технологическая (проектно-технологическая))
ОПК-5	Администрирование информационных систем, Архитектура информационных систем, Инструментальные средства информационных систем, Инфокоммуникационные системы и сети, производственная практика (технологическая (проектно-технологическая))		производственная практика (технологическая (проектно-технологическая))
ОПК-7	Администрирование информационных систем, Архитектура информационных систем, Инструментальные средства информационных систем, производственная практика (технологическая (проектно-технологическая))		производственная практика (технологическая (проектно-технологическая))

Раздел 2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Таблица 4 - Показатели оценивания компетенций на различных этапах их формирования в процессе освоения учебной дисциплины (в соответствии с РПД)

№	Раздел		
Формируемые компетенции			
	<table> <tr> <th>Показатели сформированности (в терминах «знать», «уметь», «владеть»)</th><th>Виды оценочных средств</th></tr> </table>	Показатели сформированности (в терминах «знать», «уметь», «владеть»)	Виды оценочных средств
Показатели сформированности (в терминах «знать», «уметь», «владеть»)	Виды оценочных средств		
1	Windows Presentation Foundation (WPF) и Universal Windows Platform (UWP).		
ОПК-2 ОПК-5 ОПК-7			
Знать знать современные инструментальные средства для разработки информационных систем Знать знать основы работы СУБД Знать знать основные платформы, технологии и инструментальные средства для реализации информационных систем	Проект		
Уметь уметь выбирать инструментальные средства для решения конкретных задач Уметь уметь настраивать информационные системы. Уметь уметь осуществлять выбор инструментальных средств для разработки информационных систем	Проект		
Владеть иметь навыки применения инструментальных средства при решении задач профессиональной деятельности. Владеть иметь навыки инсталляции программного обеспечения информационных систем Владеть иметь навыки владения инструментальными средствами для реализации информационных систем	Проект		
2	Мобильные приложения		
ОПК-2 ОПК-5 ОПК-7			
Знать знать современные инструментальные средства для разработки информационных систем Знать знать основы работы СУБД Знать знать основные платформы, технологии и инструментальные средства для реализации информационных систем	Проект		
Уметь уметь выбирать инструментальные средства для решения конкретных задач Уметь уметь настраивать информационные системы. Уметь уметь осуществлять выбор инструментальных средств для разработки информационных систем	Проект		
Владеть иметь навыки применения инструментальных средства при решении задач профессиональной деятельности. Владеть иметь навыки инсталляции программного обеспечения информационных систем Владеть иметь навыки владения инструментальными средствами для реализации информационных систем	Проект		

Таблица 5 - Описание уровней и критериев оценивания компетенций, описание шкал оценивания

Код	Содержание компетенции			
Уровни освоения компетенции	Содержательное описание уровня	Основные признаки выделения уровня (критерии оценки сформированности)	Пятибалльная шкала (академическая оценка)	% освоения (рейтинговая оценка)
ОПК-2	ОПК-2 способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и исп...			

ОПК-5	ОПК-5 способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем
ОПК-7	ОПК-7 способен осуществлять выбор платформ и инструментальных программно-аппаратных средств для реализации информационных систем

Раздел 3. Типовые контрольные задания и (или) иные материалы, необходимые для оценки планируемых результатов обучения по дисциплине (модулю)

1. Оценочные средства для текущего контроля

Раздел: Windows Presentation Foundation (WPF) и Universal Windows Platform (UWP).

Задания для оценки знаний

1. Проект:

Задание по проекту:

- 1) Найти программы – аналоги. Изучить типовые шаблоны, которые используются при создании продуктов данного класса. Предоставить скрины экранов.
- 2) Создать бумажный прототип будущего приложения.
- 3) Создать приложения в системе программирования

Пример проекта:

Пусть требуется создать программную систему, предназначенную для работника деканата. Такая система должна обеспечивать хранение сведений о группах и студентах, а также о результатах текущей сессии. Таким образом, для каждого студента должны храниться такие данные, как фамилия, имя, отчество студента, номер его зачетной книжки, адрес постоянной прописки и адрес, по которому студент проживает, получает или нет стипендию, а также оценки, полученные в текущей сессии, и отметки о сданных зачетах. Сведения о группе – это номер группы, факультет, кафедра, специальность, к которым она относится, год формирования группы. Работник деканата может вносить в БД следующие изменения:

- Удалить или добавить в базу студента;
- Поменять студенту номер группы, специальность, кафедру, номер зачетки;
- Занести оценки, полученные студентами на экзаменах по каждому предмету;
- По результатам сессии начислить стипендии студентам, не имеющим троек или иногородним студентам, которые имеют не более одной тройки. Право на 50 % повышение стипендии имеют студенты, получившие в сессию не более двух четверок, а на 100 % повышение – студенты, сдавшие сессию на все пятерки. Работнику деканата могут потребоваться следующие сведения:
- Студенты, обучающиеся на определенной кафедре и не сдавшие хотя бы один экзамен, с указанием группы и предмета, по которому оценка отсутствует или равна 2?
- Средний балл студентов каждой группы указанного факультета?
- Средний балл по каждому предмету?
- Список студентов указанной кафедры, которые по итогам сессии могут получать стипендию?
- Список студентов, подлежащих отчислению (не сдано более двух предметов) по всему факультету.
- Количество студентов факультета, не получающих стипендию, получающих обычную стипендию, с 50 % надбавкой и со 100 % надбавкой.
- По какому предмету больше всего неудовлетворительных оценок?
- Какой предмет студенты сдали лучше, чем другие предметы?

Необходимо предусмотреть возможность получения отчета о студентах каждой группы факультета, получающих стипендию, с указанием оценок, полученных ими, и подсчетом среднего балла каждого студента, среднего балла группы и факультета. Следует также подсчитать в группе и на факультете общее количество студентов, количество студентов, получающих обычную стипендию, получающих повышенную стипендию и не получающих стипендию. Подготовить, используя шаблон и данные из БД, приказ на отчисление студентов, не сдавших более двух предметов.

На основе примера из <https://metanit.com/sharp/entityframeworkcore/1.2.php> создать консольное приложение. Доработать его до примера <https://metanit.com/sharp/entityframeworkcore/2.1.php>

Сделать так, чтоб работало каскадное удаление. <https://metanit.com/sharp/entityframeworkcore/3.2.php>

Вариант 1

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Вариант 2

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Вариант 3

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата, код отдела.

Вариант 4

Книга. Код, название, автор, цена, число страниц, год издания, код стеллажа.

Вариант 5

Учебная дисциплина. Код, название, ФИО преподавателя, форма контроля, семестр, число часов, код специальности.

Вариант 6

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада, сумма, процент, код банка.

Вариант 7

Предмет обуви. Код, наименование, производитель, число пар, размер, цена, код отдела.

Вариант 8

Ноутбук. Код, фирма, модель, процессор, объем памяти, цена, код магазина.

Вариант 9

Билет на междугородный автобус. Код, рейс, пункт назначения, время отправления, длительность, номер места, код автобуса.

Вариант 10

Квартира. Код, дом, номер, этаж, площадь, цена, код агентства.

Вариант 11

Заказ на перевозку груза. Код, номер, дата, адрес доставки, вес гру-

за, стоимость перевозки, код водителя.

Вариант 12

Спортсмен. Код, ФИО, вид спорта, дата рождения, пол, рост, вес, код тренера.

Вариант 13

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Вариант 14

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Вариант 15

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата, код отдела.

Вариант 16

Книга. Код, название, автор, цена, число страниц, год издания, код стеллажа.

Вариант 17

Учебная дисциплина. Код, название, ФИО преподавателя, форма контроля, семестр, число часов, код специальности.

Вариант 18

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада, сумма, процент, код банка.

Изучить «Создание моделей в Entity Framework Core» <https://metanit.com/sharp/entityframeworkcore/2.3.php>.
Изучить «Отношения между моделями»

<https://metanit.com/sharp/entityframeworkcore/3.1.php>

Создать консольное приложение, работающее с взаимосвязанными таблицами БД.

Сделать так, чтоб работало каскадное удаление. <https://metanit.com/sharp/entityframeworkcore/3.2.php>

Вариант 1

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Научный руководитель. Код, ФИО, должность, фото.

Вариант 2

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Магазин. Код, название, телефон, фото.

Вариант 3

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата, код отдела.

Отдел. Код, название, телефон, фото.

Вариант 4

Книга. Код, название, автор, цена, число страниц, год издания, код стеллажа.

Стеллаж. Код, число полок, название отдела, фото.

Вариант 5

Учебная дисциплина. Код, название, ФИО преподавателя, форма контроля, семестр, число часов, код специальности.

Специальность. Код, название, факультет, фото.

Вариант 6

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада, сумма, процент, код банка.

Банк. Код, название, адрес центрального офиса, фото.

Вариант 7

Предмет обуви. Код, наименование, производитель, число пар, размер, цена, код отдела.

Отдел. Код, название, число сотрудников, фото.

Вариант 8

Ноутбук. Код, фирма, модель, процессор, объем памяти, цена, код магазина.

Магазин. Код, название, адрес, фото.

Вариант 9

Билет на междугородный автобус. Код, рейс, пункт назначения, время отправления, длительность, номер места, код автобуса.

Автобус. Код, модель, число посадочных мест, фото.

Вариант 10

Квартира. Код, дом, номер, этаж, площадь, цена, код агентства.

Агентство недвижимости. Код, название, телефон, фото.

Вариант 11

Заказ на перевозку груза. Код, номер, дата, адрес доставки, вес груза, стоимость перевозки, код водителя.

Водитель. Код, ФИО, дата рождения, фото.

Вариант 12

Спортсмен. Код, ФИО, вид спорта, дата рождения, пол, рост, вес, код тренера.

Тренер. Код, ФИО, звание, фото.

Вариант 13

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Научный руководитель. Код, ФИО, должность, фото.

Вариант 14

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Магазин. Код, название, телефон, фото.

Вариант 15

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата, код отдела.

Отдел. Код, название, телефон, фото.

Вариант 16

Книга. Код, название, автор, цена, число страниц, год издания, код стеллажа.

Стеллаж. Код, число полок, название отдела, фото.

Вариант 17

Учебная дисциплина. Код, название, ФИО преподавателя, форма контроля, семестр, число часов, код специальности.

Специальность. Код, название, факультет, фото.

Вариант 18

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада, сумма, процент, код банка.

Банк. Код, название, адрес центрального офиса, фото.

На основе примера из лекции создать приложение на основе страничной организации приложения, создания меню, панели команд, табличных элементов управления и системы команд для выполнения задач приложения. Приложение должно выглядеть, как в лекции.

Задания для оценки умений

1. Проект:

Задание по проекту:

- 1) Найти программы – аналоги. Изучить типовые шаблоны, которые используются при создании продуктов данного класса. Предоставить скрины экранов.
- 2) Создать бумажный прототип будущего приложения.
- 3) Создать приложения в системе программирования

Пример проекта:

Пусть требуется создать программную систему, предназначенную для работника деканата. Такая система должна обеспечивать хранение сведений о группах и студентах, а также о результатах текущей сессии. Таким образом, для каждого студента должны храниться такие данные, как фамилия, имя, отчество студента, номер его зачетной книжки, адрес постоянной прописки и адрес, по которому студент проживает, получает или нет стипендию, а также оценки, полученные в текущей сессии, и отметки о сданных зачетах. Сведения о группе – это номер группы, факультет, кафедра, специальность, к которым она относится, год формирования группы. Работник деканата может вносить в БД следующие изменения:

- Удалить или добавить в базу студента;
- Поменять студенту номер группы, специальность, кафедру, номер зачетки;
- Занести оценки, полученные студентами на экзаменах по каждому предмету;
- По результатам сессии начислить стипендии студентам, не имеющим троек или иногородним студентам, которые имеют не более одной тройки. Право на 50 % повышение стипендии имеют студенты, получившие в сессию не более двух четверок, а на 100 % повышение – студенты, сдавшие сессию на все пятерки. Работнику деканата могут потребоваться следующие сведения:
 - Студенты, обучающиеся на определенной кафедре и не сдавшие хотя бы один экзамен, с указанием группы и предмета, по которому оценка отсутствует или равна 2?
 - Средний балл студентов каждой группы указанного факультета?
 - Средний балл по каждому предмету?
 - Список студентов указанной кафедры, которые по итогам сессии могут получать стипендию?
 - Список студентов, подлежащих отчислению (не сдано более двух предметов) по всему факультету.
 - Количество студентов факультета, не получающих стипендию, получающих обычную стипендию, с 50 % надбавкой и со 100 % надбавкой.
 - По какому предмету больше всего неудовлетворительных оценок?
 - Какой предмет студенты сдали лучше, чем другие предметы?

Необходимо предусмотреть возможность получения отчета о студентах каждой группы факультета, получающих стипендию, с указанием оценок, полученных ими, и подсчетом среднего балла каждого студента, среднего балла группы и факультета. Следует также подсчитать в группе и на факультете общее количество студентов, количество студентов, получающих обычную стипендию, получающих повышенную стипендию и не получающих стипендию. Подготовить, используя шаблон и данные из БД, приказ на отчисление студентов, не сдавших более двух предметов.

На основе примера из <https://metanit.com/sharp/entityframeworkcore/1.2.php> создать консольное приложение. Доработать его до примера <https://metanit.com/sharp/entityframeworkcore/2.1.php>

Сделать так, чтоб работало каскадное удаление. <https://metanit.com/sharp/entityframeworkcore/3.2.php>

Вариант 1

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Вариант 2

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Вариант 3

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата, код отдела.

Вариант 4

Книга. Код, название, автор, цена, число страниц, год издания, код стеллажа.

Вариант 5

Учебная дисциплина. Код, название, ФИО преподавателя, форма контроля, семестр, число часов, код специальности.

Вариант 6

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада, сумма, процент, код банка.

Вариант 7

Предмет обуви. Код, наименование, производитель, число пар, размер, цена, код отдела.

Вариант 8

Ноутбук. Код, фирма, модель, процессор, объем памяти, цена, код магазина.

Вариант 9

Билет на междугородный автобус. Код, рейс, пункт назначения, время отправления, длительность, номер места, код автобуса.

Вариант 10

Квартира. Код, дом, номер, этаж, площадь, цена, код агентства.

Вариант 11

Заказ на перевозку груза. Код, номер, дата, адрес доставки, вес груза, стоимость перевозки, код водителя.

Вариант 12

Спортсмен. Код, ФИО, вид спорта, дата рождения, пол, рост, вес, код тренера.

Вариант 13

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Вариант 14

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Вариант 15

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата, код отдела.

Вариант 16

Книга. Код, название, автор, цена, число страниц, год издания, код стеллажа.

Вариант 17

Учебная дисциплина. Код, название, ФИО преподавателя, форма контроля, семестр, число часов, код специальности.

Вариант 18

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада, сумма, процент, код банка.

Изучить «Создание моделей в Entity Framework Core» <https://metanit.com/sharp/entityframeworkcore/2.3.php>.
Изучить «Отношения между моделями»

<https://metanit.com/sharp/entityframeworkcore/3.1.php>

Создать консольное приложение, работающее с взаимосвязанными таблицами БД.

Сделать так, чтоб работало каскадное удаление. <https://metanit.com/sharp/entityframeworkcore/3.2.php>

Вариант 1

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Научный руководитель. Код, ФИО, должность, фото.

Вариант 2

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Магазин. Код, название, телефон, фото.

Вариант 3

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата, код отдела.

Отдел. Код, название, телефон, фото.

Вариант 4

Книга. Код, название, автор, цена, число страниц, год издания, код

стеллажа.

Стеллаж. Код, число полок, название отдела, фото.

Вариант 5

Учебная дисциплина. Код, название, ФИО преподавателя, форма контроля, семестр, число часов, код специальности.

Специальность. Код, название, факультет, фото.

Вариант 6

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада, сумма, процент, код банка.

Банк. Код, название, адрес центрального офиса, фото.

Вариант 7

Предмет обуви. Код, наименование, производитель, число пар, размер, цена, код отдела.

Отдел. Код, название, число сотрудников, фото.

Вариант 8

Ноутбук. Код, фирма, модель, процессор, объем памяти, цена, код магазина.

Магазин. Код, название, адрес, фото.

Вариант 9

Билет на междугородный автобус. Код, рейс, пункт назначения, время отправления, длительность, номер места, код автобуса.

Автобус. Код, модель, число посадочных мест, фото.

Вариант 10

Квартира. Код, дом, номер, этаж, площадь, цена, код агентства.

Агентство недвижимости. Код, название, телефон, фото.

Вариант 11

Заказ на перевозку груза. Код, номер, дата, адрес доставки, вес груза, стоимость перевозки, код водителя.

Водитель. Код, ФИО, дата рождения, фото.

Вариант 12

Спортсмен. Код, ФИО, вид спорта, дата рождения, пол, рост, вес, код тренера.

Тренер. Код, ФИО, звание, фото.

Вариант 13

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Научный руководитель. Код, ФИО, должность, фото.

Вариант 14

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Магазин. Код, название, телефон, фото.

Вариант 15

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата, код отдела.

Отдел. Код, название, телефон, фото.

Вариант 16

Книга. Код, название, автор, цена, число страниц, год издания, код стеллажа.

Стеллаж. Код, число полок, название отдела, фото.

Вариант 17

Учебная дисциплина. Код, название, ФИО преподавателя, форма контроля, семестр, число часов, код специальности.

Специальность. Код, название, факультет, фото.

Вариант 18

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада, сумма, процент, код банка.

Банк. Код, название, адрес центрального офиса, фото.

На основе примера из лекции создать приложение на основе страничной организации приложения, создания меню, панели команд, табличных элементов управления и системы команд для выполнения задач приложения. Приложение должно выглядеть, как в лекции.

Задания для оценки владений

1. Проект:

Задание по проекту:

- 1) Найти программы – аналоги. Изучить типовые шаблоны, которые используются при создании продуктов данного класса. Предоставить скрины экранов.
- 2) Создать бумажный прототип будущего приложения.
- 3) Создать приложения в системе программирования

Пример проекта:

Пусть требуется создать программную систему, предназначенную для работника деканата. Такая система должна обеспечивать хранение сведений о группах и студентах, а также о результатах текущей сессии. Таким образом, для каждого студента должны храниться такие данные, как фамилия, имя, отчество студента, номер его зачетной книжки, адрес постоянной прописки и адрес, по которому студент проживает, получает или нет стипендию, а также оценки, полученные в текущей сессии, и отметки о сданных зачетах. Сведения о группе – это номер группы, факультет, кафедра, специальность, к которой она относится, год формирования группы. Работник деканата может вносить в БД следующие изменения:

- Удалить или добавить в базу студента;
- Поменять студенту номер группы, специальность, кафедру, номер зачетки;
- Занести оценки, полученные студентами на экзаменах по каждому предмету;
- По результатам сессии начислить стипендии студентам, не имеющим троек или иногородним студентам, которые имеют не более одной тройки. Право на 50 % повышение стипендии имеют студенты, получившие в сессии не более двух четверок, а на 100 % повышение – студенты, сдавшие сессию на все пятерки. Работнику деканата могут потребоваться следующие сведения:
- Студенты, обучающиеся на определенной кафедре и не сдавшие хотя бы один экзамен, с указанием группы и предмета, по которому оценка отсутствует или равна 2?
- Средний балл студентов каждой группы указанного факультета?
- Средний балл по каждому предмету?
- Список студентов указанной кафедры, которые по итогам сессии могут получать стипендию?
- Список студентов, подлежащих отчислению (не сдано более двух предметов) по всему факультету.
- Количество студентов факультета, не получающих стипендию, получающих обычную стипендию, с 50 % надбавкой и со 100 % надбавкой.
- По какому предмету больше всего неудовлетворительных оценок?
- Какой предмет студенты сдали лучше, чем другие предметы?

Необходимо предусмотреть возможность получения отчета о студентах каждой группы факультета, получающих стипендию, с указанием оценок, полученных ими, и подсчетом среднего балла каждого студента, среднего балла группы и факультета. Следует также подсчитать в группе и на факультете общее количество студентов, количество студентов, получающих обычную стипендию, получающих повышенную стипендию и не получающих стипендию. Подготовить, используя шаблон и данные из БД, приказ на отчисление студентов, не сдавших более двух предметов.

На основе примера из <https://metanit.com/sharp/entityframeworkcore/1.2.php> создать консольное приложение. Доработать его до примера <https://metanit.com/sharp/entityframeworkcore/2.1.php>

Сделать так, чтоб работало каскадное удаление. <https://metanit.com/sharp/entityframeworkcore/3.2.php>

Вариант 1

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Вариант 2

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Вариант 3

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата, код отдела.

Вариант 4

Книга. Код, название, автор, цена, число страниц, год издания, код стеллажа.

Вариант 5

Учебная дисциплина. Код, название, ФИО преподавателя, форма контроля, семестр, число часов, код специальности.

Вариант 6

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада, сумма, процент, код банка.

Вариант 7

Предмет обуви. Код, наименование, производитель, число пар, размер, цена, код отдела.

Вариант 8

Ноутбук. Код, фирма, модель, процессор, объем памяти, цена, код магазина.

Вариант 9

Билет на междугородный автобус. Код, рейс, пункт назначения,

время отправления, длительность, номер места, код автобуса.

Вариант 10

Квартира. Код, дом, номер, этаж, площадь, цена, код агентства.

Вариант 11

Заказ на перевозку груза. Код, номер, дата, адрес доставки, вес груза, стоимость перевозки, код водителя.

Вариант 12

Спортсмен. Код, ФИО, вид спорта, дата рождения, пол, рост, вес, код тренера.

Вариант 13

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Вариант 14

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Вариант 15

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата, код отдела.

Вариант 16

Книга. Код, название, автор, цена, число страниц, год издания, код стеллажа.

Вариант 17

Учебная дисциплина. Код, название, ФИО преподавателя, форма контроля, семестр, число часов, код специальности.

Вариант 18

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада, сумма, процент, код банка.

Изучить «Создание моделей в Entity Framework Core» <https://metanit.com/sharp/entityframeworkcore/2.3.php>.
Изучить «Отношения между моделями»

<https://metanit.com/sharp/entityframeworkcore/3.1.php>

Создать консольное приложение, работающее с взаимосвязанными таблицами БД.

Сделать так, чтоб работало каскадное удаление. <https://metanit.com/sharp/entityframeworkcore/3.2.php>

Вариант 1

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Научный руководитель. Код, ФИО, должность, фото.

Вариант 2

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Магазин. Код, название, телефон, фото.

Вариант 3

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата, код отдела.

Отдел. Код, название, телефон, фото.

Вариант 4

Книга. Код, название, автор, цена, число страниц, год издания, код стеллажа.

Стеллаж. Код, число полок, название отдела, фото.

Вариант 5

Учебная дисциплина. Код, название, ФИО преподавателя, форма контроля, семестр, число часов, код специальности.

Специальность. Код, название, факультет, фото.

Вариант 6

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада, сумма, процент, код банка.

Банк. Код, название, адрес центрального офиса, фото.

Вариант 7

Предмет обуви. Код, наименование, производитель, число пар, размер, цена, код отдела.

Отдел. Код, название, число сотрудников, фото.

Вариант 8

Ноутбук. Код, фирма, модель, процессор, объем памяти, цена, код магазина.

Магазин. Код, название, адрес, фото.

Вариант 9

Билет на междугородный автобус. Код, рейс, пункт назначения, время отправления, длительность, номер места, код автобуса.

Автобус. Код, модель, число посадочных мест, фото.

Вариант 10

Квартира. Код, дом, номер, этаж, площадь, цена, код агентства.

Агентство недвижимости. Код, название, телефон, фото.

Вариант 11

Заказ на перевозку груза. Код, номер, дата, адрес доставки, вес груза, стоимость перевозки, код водителя.

Водитель. Код, ФИО, дата рождения, фото.

Вариант 12

Спортсмен. Код, ФИО, вид спорта, дата рождения, пол, рост, вес, код тренера.

Тренер. Код, ФИО, звание, фото.

Вариант 13

Студент. Код, ФИО, группа, пол, дата рождения, средний бал, код научного руководителя.

Научный руководитель. Код, ФИО, должность, фото.

Вариант 14

Автомобиль. Код, фирма, модель, год выпуска, цена, расход топлива, код магазина.

Магазин. Код, название, телефон, фото.

Вариант 15

Сотрудник. Код, ФИО, пол, дата рождения, должность, зарплата,
код отдела.

Отдел. Код, название, телефон, фото.

Вариант 16

Книга. Код, название, автор, цена, число страниц, год издания, код
стеллажа.

Стеллаж. Код, число полок, название отдела, фото.

Вариант 17

Учебная дисциплина. Код, название, ФИО преподавателя, форма
контроля, семестр, число часов, код специальности.

Специальность. Код, название, факультет, фото.

Вариант 18

Банковский вклад. Код, номер счета, ФИО вкладчика, дата вклада,
сумма, процент, код банка.

Банк. Код, название, адрес центрального офиса, фото.

На основе примера из лекции создать приложение на основе страничной организации приложения, создания меню, панели команд, табличных элементов управления и системы команд для выполнения задач приложения. Приложение должно выглядеть, как в лекции.

Раздел: Мобильные приложения

Задания для оценки знаний

1. Проект:

Требования к мобильному приложению

Функциональные требования:

- 1) пользователь должен иметь возможность вводить позиции вручную;
- 2) программа должна работать с фото путем загрузки изображения с устройства пользователя;
- 3) программа должна предоставлять функцию ввода количества ;

- 4) пользователь должен иметь возможность выбора и отмены выбора пункта из распознанного/введенного списка;
- 5) программа должна выводить выбранные объекты, их количество и сумму к оплате для каждого пользователя;
- 6) программа должна выводить на экран предупреждение, сообщающее о несовпадении реальной итоговой суммы и суммы по пользователям.

Нефункциональные требования:

- 1) приложение должно быть разработано для ОС Android (версия Lollipop и более поздние);
- 2) распознавание данных должно осуществляться с помощью открытой библиотеки OpenCV;
- 3) приложение должно работать в автономном режиме (без подключения к сети Интернет);
- 5) финансовые расчеты в приложении должны производиться в рублевой валюте;
- 6) интерфейс приложения должен быть на русском языке

Задания по вариантам::

Создать мобильное приложение, работающее с вашей базой данных: редактирование, добавление, удаление данных.

Вид страховки (Выпадающий список Picker), Фιο страхуемого (3 текстовых поля), срок страховки (дата), сумма (Slider или Stepper), фото страхуемого (картинка).

Объект недвижимости (Выпадающий список Picker), фιο владельца (3 текстовых поля), адрес, стоимость (Slider или Stepper), дата просмотра (дата), фото объекта (картинка).

Фιο студента (3 текстовых поля), номер группы (Выпадающий список Picker), дата рождения (дата), размер стипендии (Slider или Stepper), фото (картинка),

Фιο сотрудника (3 текстовых поля), должность (Выпадающий список Picker), отдел, дата рождения (дата), размер оклада (Slider или Stepper), фото (картинка)

Марка автомобиля (Выпадающий список Picker), год выпуска (дата), цвет фото (картинка), стоимость (Slider или Stepper), фιο владельца (3 текстовых поля)

Список номеров гостиницы (Выпадающий список Picker), фιο постояльца (3 текстовых поля), дата заезда (дата), количество дней (Slider или Stepper), фото номера (картинка)

Список рейсов самолётов (Выпадающий список Picker), дата и время вылета (дата), стоимость билета (Slider или Stepper), количество мест (Slider или Stepper), фιο пассажира (3 текстовых поля), фото (картинка)

Список предметов в ВУЗе (Выпадающий список Picker), фιο преподавателя (3 текстовых поля), дата экзамена (дата), количество сдающих экзамен (Slider или Stepper), фото ведомости (картинка)

Тип турпутевки (Выпадающий список Picker), город (Выпадающий список Picker), стоимость (Slider или Stepper), дата отправления (дата), количество дней (Slider или Stepper), фото отеля (картинка), фιο отдыхающего (3 текстовых поля)

Тип товаров в магазине (Выпадающий список Picker), наименование товара (текстовое поле), цена (Slider или Stepper), количество (Slider или Stepper), срок годности (дата), фото товара (картинка)

Тип комплектующих ПК (Выпадающий список Picker), наименование (текстовое поле), цена (Slider или Stepper), производитель, дата поступления в магазин (дата), фото товара (картинка)

Список моделей телефонов (Выпадающий список Picker), наименование (текстовое поле), цвет, цена (Slider или Stepper), количество (Slider или Stepper), дата поступления в магазин (дата), фото (картинка)

Название видеофильма (текстовое поле), жанр (Выпадающий список Picker), длительность, цена диска (Slider или Stepper), страна (текстовое поле), дата выхода на экран (дата), рекламное фото (картинка)

Список наименований врачей (Выпадающий список Picker), фιο врача (3 текстовых поля), цена приема (Slider или Stepper), дата приема (дата), фото врача (картинка)

Тип печатных изданий (Выпадающий список Picker), название журнала или газеты (текстовое поле), дата выхода (дата), цена (Slider или Stepper), количество (Slider или Stepper), фото журнала (картинка)

тип тарифа сотовой связи (Выпадающий список Picker), название тарифа (текстовое поле), цена (Slider или Stepper), параметры тарифа (несколько Slider или Stepper), дата включения тарифа (дата), реклама тарифа фото (картинка)

Список кредитов в банке (Выпадающий список Picker), сумма кредита (Slider или Stepper), процентная ставка (Slider или Stepper), ф.и.о. клиента (3 текстовых поля), дата платежа по кредиту (дата), фото клиента (картинка)

Список фильмов в кинотеатре (текстовое поле), наименование фильма, жанр (Выпадающий список Picker), возрастные ограничения (Slider или Stepper), цена билета (Slider или Stepper), дата премьеры (дата), фото (картинка).

Задания для оценки умений

1. Проект:

Требования к мобильному приложению

Функциональные требования:

- 1) пользователь должен иметь возможность вводить позиции вручную;
- 2) программа должна работать с фото путем загрузки изображения с устройства пользователя;
- 3) программа должна предоставлять функцию ввода количества;
- 4) пользователь должен иметь возможность выбора и отмены выбора пункта из распознанного/введенного списка;
- 5) программа должна выводить выбранные объекты, их количество и сумму к оплате для каждого пользователя;
- 6) программа должна выводить на экран предупреждение, сообщающее о несовпадении реальной итоговой суммы и суммы по пользователям.

Нефункциональные требования:

- 1) приложение должно быть разработано для ОС Android (версия Lollipop и более поздние);
- 2) распознавание данных должно осуществляться с помощью открытой библиотеки OpenCV;
- 3) приложение должно работать в автономном режиме (без подключения к сети Интернет);
- 5) финансовые расчеты в приложении должны производиться в рублевой валюте;
- 6) интерфейс приложения должен быть на русском языке

Задания по вариантам:

Создать мобильное приложение, работающее с вашей базой данных: редактирование, добавление, удаление данных.

Вид страховки (Выпадающий список Picker), Ф.и.о. страхуемого (3 текстовых поля), срок страховки (дата), сумма (Slider или Stepper), фото страхуемого (картинка).

Объект недвижимости (Выпадающий список Picker), ф.и.о. владельца (3 текстовых поля), адрес, стоимость (Slider или Stepper), дата просмотра (дата), фото объекта (картинка).

Ф.и.о. студента (3 текстовых поля), номер группы (Выпадающий список Picker), дата рождения (дата), размер стипендии (Slider или Stepper), фото (картинка),

Ф.и.о. сотрудника (3 текстовых поля), должность (Выпадающий список Picker), отдел, дата рождения (дата), размер оклада (Slider или Stepper), фото (картинка)

Марка автомобиля (Выпадающий список Picker), год выпуска (дата), цвет фото (картинка), стоимость (Slider или Stepper), ф.и.о. владельца (3 текстовых поля)

Список номеров гостиницы (Выпадающий список Picker), ф.и.о. постояльца (3 текстовых поля), дата заезда (дата), количество дней (Slider или Stepper), фото номера (картинка)

Список рейсов самолётов (Выпадающий список Picker), дата и время вылета (дата), стоимость билета (Slider или Stepper), количество мест (Slider или Stepper), ф.и.о. пассажира (3 текстовых поля), фото (картинка)

Список предметов в ВУЗе(Выпадающий список Picker), фии преподавателя(3 текстовых поля), дата экзамена(дата), количество сдающих экзамен (Slider или Stepper), фото ведомости (картинка)

Тип турпутевки(Выпадающий список Picker), город(Выпадающий список Picker), стоимость(Slider или Stepper), дата отправления (дата), количество дней(Slider или Stepper), фото отеля(картинка), фии отдыхающего (3 текстовых поля)

Тип товаров в магазине(Выпадающий список Picker), наименование товара (текстовое поле), цена(Slider или Stepper), количество(Slider или Stepper), срок годности(дата), фото товара (картинка)

Тип комплектующих ПК (Выпадающий список Picker), наименование (текстовое поле), цена(Slider или Stepper), производитель, дата поступления в магазин (дата), фото товара(картинка)

Список моделей телефонов (Выпадающий список Picker), наименование (текстовое поле),цвет, цена(Slider или Stepper), количество(Slider или Stepper), дата поступления в магазин (дата), фото (картинка)

Название видеофильма (текстовое поле), жанр (Выпадающий список Picker), длительность, цена диска(Slider или Stepper), страна (текстовое поле), дата выхода на экран (дата), рекламное фото (картинка)

Список наименований врачей (Выпадающий список Picker), фии врача (3 текстовых поля), цена приема(Slider или Stepper), дата приема(дата), фото врача (картинка)

Тип печатных изданий (Выпадающий список Picker), название журнала или газеты (текстовое поле), дата выхода (дата), цена(Slider или Stepper), количество(Slider или Stepper), фото журнала (картинка)

тип тарифа сотовой связи (Выпадающий список Picker), название тарифа (текстовое поле), цена(Slider или Stepper), параметры тарифа (несколько Slider или Stepper), дата включения тарифа(дата), реклама тарифа фото (картинка)

Список кредитов в банке (Выпадающий список Picker), сумма кредита(Slider или Stepper), процентная ставка(Slider или Stepper), фии клиента (3 текстовых поля), дата платежа по кредиту (дата), фото клиента (картинка)

Список фильмов в кинотеатре (текстовое поле), наименование фильма, жанр(Выпадающий список Picker), возрастные ограничения(Slider или Stepper), цена билета(Slider или Stepper), дата премьеры(дата), фото (картинка).

Задания для оценки владений

1. Проект:

Требования к мобильному приложению

Функциональные требования:

- 1) пользователь должен иметь возможность вводить позиции вручную;
- 2) программа должна работать с фото путем загрузки изображения с устройства пользователя;
- 3) программа должна предоставлять функцию ввода количества ;
- 4) пользователь должен иметь возможность выбора и отмены выбора пункта из распознанного/введенного списка;
- 5) программа должна выводить выбранные объекты, их количество и сумму к оплате для каждого пользователя;
- 6) программа должна выводить на экран предупреждение, сообщающее о несовпадении реальной итоговой суммы и суммы по пользователям.

Нефункциональные требования:

- 1) приложение должно быть разработано для ОС Android (версия Lollipop и более поздние);
- 2) распознавание данных должно осуществляться с помощью открытой библиотеки OpenCV;
- 3) приложение должно работать в автономном режиме (без подключения к сети Интернет);
- 5) финансовые расчеты в приложении должны производиться в рублевой валюте;
- 6) интерфейс приложения должен быть на русском языке

Задания по вариантам::

Создать мобильное приложение, работающее с вашей базой данных: редактирование, добавление, удаление данных.

Вид страховки (Выпадающий список Picker), Фιο страховемого (3 текстовых поля), срок страховки (дата), сумма (Slider или Stepper), фото страховемого (картинка).

Объект недвижимости (Выпадающий список Picker), фιο владельца (3 текстовых поля), адрес, стоимость (Slider или Stepper), дата просмотра (дата), фото объекта (картинка).

Фιο студента (3 текстовых поля), номер группы (Выпадающий список Picker), дата рождения (дата), размер стипендии (Slider или Stepper), фото (картинка),

Фιο сотрудника (3 текстовых поля), должность (Выпадающий список Picker), отдел, дата рождения (дата), размер оклада (Slider или Stepper), фото (картинка)

Марка автомобиля (Выпадающий список Picker), год выпуска (дата), цвет фото (картинка), стоимость (Slider или Stepper), фιο владельца (3 текстовых поля)

Список номеров гостиницы (Выпадающий список Picker), фιο постояльца (3 текстовых поля), дата заезда (дата), количество дней (Slider или Stepper), фото номера (картинка)

Список рейсов самолётов (Выпадающий список Picker), дата и время вылета (дата), стоимость билета (Slider или Stepper), количество мест (Slider или Stepper), фιο пассажира (3 текстовых поля), фото (картинка)

Список предметов в ВУЗе (Выпадающий список Picker), фιο преподавателя (3 текстовых поля), дата экзамена (дата), количество сдающих экзамен (Slider или Stepper), фото ведомости (картинка)

Тип турпутевки (Выпадающий список Picker), город (Выпадающий список Picker), стоимость (Slider или Stepper), дата отправления (дата), количество дней (Slider или Stepper), фото отеля (картинка), фιο отдыхающего (3 текстовых поля)

Тип товаров в магазине (Выпадающий список Picker), наименование товара (текстовое поле), цена (Slider или Stepper), количество (Slider или Stepper), срок годности (дата), фото товара (картинка)

Тип комплектующих ПК (Выпадающий список Picker), наименование (текстовое поле), цена (Slider или Stepper), производитель, дата поступления в магазин (дата), фото товара (картинка)

Список моделей телефонов (Выпадающий список Picker), наименование (текстовое поле), цвет, цена (Slider или Stepper), количество (Slider или Stepper), дата поступления в магазин (дата), фото (картинка)

Название видеофильма (текстовое поле), жанр (Выпадающий список Picker), длительность, цена диска (Slider или Stepper), страна (текстовое поле), дата выхода на экран (дата), рекламное фото (картинка)

Список наименований врачей (Выпадающий список Picker), фιο врача (3 текстовых поля), цена приема (Slider или Stepper), дата приема (дата), фото врача (картинка)

Тип печатных изданий (Выпадающий список Picker), название журнала или газеты (текстовое поле), дата выхода (дата), цена (Slider или Stepper), количество (Slider или Stepper), фото журнала (картинка)

тип тарифа сотовой связи (Выпадающий список Picker), название тарифа (текстовое поле), цена (Slider или Stepper), параметры тарифа (несколько Slider или Stepper), дата включения тарифа (дата), реклама тарифа фото (картинка)

Список кредитов в банке (Выпадающий список Picker), сумма кредита (Slider или Stepper), процентная ставка (Slider или Stepper), фιο клиента (3 текстовых поля), дата платежа по кредиту (дата), фото клиента (картинка)

Список фильмов в кинотеатре (текстовое поле), наименование фильма, жанр (Выпадающий список Picker), возрастные ограничения (Slider или Stepper), цена билета (Slider или Stepper), дата премьеры (дата), фото (картинка).

2. Оценочные средства для промежуточной аттестации

1. Зачет

Вопросы к зачету:

1. Преимущества технологии WPF
2. Назначение XAML
3. Код программной части
4. Элементы управления
5. Макет приложения
6. Основы графики
7. XAML декларативный язык разметки.
8. описание синтаксиса языка XAML
9. Свойства содержимого XAML
10. Корневые элементы XAML и пространства имен XAML
11. События и код программной части XAML
12. Инструментальное средство для разработки информационной системы
13. Виды инструментальных средств для разработки web-ориентированных информационных систем.
14. Инструментальные системы для работы с базами данных
15. Xamarin и кросс-платформенная разработка. Введение в Xamarin.
16. Создание и настройка проекта
17. Создание приложения
18. Создание графического интерфейса XAML
19. Взаимодействие XAML и C#
20. Метод LoadFromXaml и загрузка XAML
21. Расширения разметки XAML
22. Элементы компоновки
23. Элементы в Xamarin и их свойства
24. Основы навигации
25. стек навигации
26. Передача данных при навигации
27. Сообщения и MessagingCenter
28. Основные операции с SQLite
29. Асинхронное подключение к SQLite
30. Подключение к существующей базе данных
31. Подключение к сети
32. Класс HttpClient и отправка запросов
33. Получение данных с сервера в json
34. Создание веб-сервиса
35. Взаимодействие с веб-сервисом
36. Создание интерфейса для работы с веб-сервисом

Раздел 4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

1. Для текущего контроля используются следующие оценочные средства:

1. Проект

Проект – это самостоятельное, развёрнутое решение обучающимся, или группой обучающихся какой-либо проблемы научно-исследовательского, творческого или практического характера.

Этапы в создании проектов.

1. Выбор проблемы.
2. Постановка целей.
3. Постановка задач (подцелей).
4. Информационная подготовка.
5. Образование творческих групп (по желанию).
6. Внутригрупповая или индивидуальная работа.
7. Внутригрупповая дискуссия.
8. Общественная презентация – защита проекта.

2. Описание процедуры промежуточной аттестации

Оценка за зачет/экзамен может быть выставлена по результатам текущего рейтинга. Текущий рейтинг – это результаты выполнения практических работ в ходе обучения, контрольных работ, выполнения заданий к лекциям (при наличии) и др. видов заданий.

Результаты текущего рейтинга доводятся до студентов до начала экзаменационной сессии.

Цель зачета – проверка и оценка уровня полученных студентом специальных знаний по учебной дисциплине и соответствующих им умений и навыков, а также умения логически мыслить, аргументировать избранную научную позицию, реагировать на дополнительные вопросы, ориентироваться в массиве информации.

Зачет может проводиться как в формате, аналогичном проведению экзамена, так и в других формах, основанных на выполнении индивидуального или группового задания, позволяющего осуществить контроль знаний и полученных навыков.

Подготовка к зачету начинается с первого занятия по дисциплине, на котором обучающиеся получают предварительный перечень вопросов к зачёту и список рекомендуемой литературы, их ставят в известность относительно критериев выставления зачёта и специфике текущей и итоговой аттестации. С самого начала желательно планомерно осваивать материал, руководствуясь перечнем вопросов к зачету и списком рекомендуемой литературы, а также путём самостоятельного конспектирования материалов занятий и результатов самостоятельного изучения учебных вопросов.

По результатам сдачи зачета выставляется оценка «зачтено» или «не зачтено».