

УТВЪРДИЛ

Директор:.....

Разработил:

Разработил:

**ГОДИШНО ТЕМАТИЧНО РАЗПРЕДЕЛЕНИЕ
ПО ИНФОРМАТИКА**

за 12. клас профилирана подготовка

МОДУЛ 4. „ПРОГРАМИРАНЕ НА ИНФОРМАЦИОННИ СИСТЕМИ СЪС С#“

УЧЕБНО СЪДЪРЖАНИЕ Втори учебен срок – 13 седмици по 4 часа седмично = 52 часа

Теми	Компетентности като очаквани резултати от обучението	Нови понятия
1. Разработване на Информационни системи		
1.1 Конфигуриране на работна среда за програмиране на приложения за СУБД	<ul style="list-style-type: none">• Прилага стандартни средства за стартиране на сървър• на база от данни• Подбира доставчици на бази от данни• Конфигурира източник на данни и създава връзка към релационна база от данни с графични средства на интегрирана среда за програмиране• Описва основните роли на групите потребители на• СУБД	<ul style="list-style-type: none">• Сървърна база от данни, Клиентско• приложение на БД,• Доставчик на база от данни,• Текстово описание на връзката(connection string)• Източник на данни

1.2 Транзакции на операции с БД	<ul style="list-style-type: none"> • Обяснява характерни особености на изпълнение на транзакции при паралелен достъп до данни в СУБД • Създава примерна база от данни и таблици с графични средства на интегрирана среда за програмиране • Изпълнява транзакции при въвеждане и редактиране на данни в таблица с графични средства на интегрирана среда за програмиране • Архивира и възстановява база от данни, посредством стандартен език за описване на транзакции • Разбира необходимостта от защита и сигурност на данните 	<ul style="list-style-type: none"> • Паралелен достъп до данни • Транзакция, обхват на транзакция, свойства (ACID) на транзакция, записване на промени от транзакция (COMMIT) и възстановяване на промени от транзакция (ROLLBACK)
1.3 Обектно ориентиран модел на релационна БД	<ul style="list-style-type: none"> • Описва характерните особености и компонентите на обектно ориентиран модел на база от данни • Назовава предимства и недостатъци в приложението на обектно ориентиран модел на база от данни • Генерира обектно ориентиран модел на зададена база от данни с графични средства на интегрирана среда за програмиране • Идентифицира характерни елементи в структурата на генерираните класове от обектно ориентиран модел 	<ul style="list-style-type: none"> • обектно ориентиран модел на база от данни • Моделиране на основен ключ • Свойства на данни (get/set) • Клас за представяне на категория (entity) от данни • Клас за управление на категории (entity) от данни
1.4 Моделиране на графичен интерфейс за ИС	<ul style="list-style-type: none"> • Изготвя програмна реализация на графичен потребителски интерфейс с помощни графични средства. • Моделира графични компоненти (етикет, текстово поле, текстова област, списък, падащ списък, полета за избор на опция, бутони) и разположението им в графичен прозорец като прилага добри практики за програмиране • Програмира събития, породени от натискане на бутон, избор на елемент от списък и на поле с опция • Създава софтуерно приложение за обработка на данни, въведени с графичен потребителски интерфейс и извеждане на получените резултати 	<ul style="list-style-type: none"> •

1.5 Графично приложение на обектно ориентиран модел на БД	<ul style="list-style-type: none"> • Генерира софтуерно приложение за извеждане на записите на таблица от база от данни в таблица на графичен прозорец като се използват графични помощни средства • Свързва графични компоненти (текстово поле, текстова област, списък, падащ списък) към полета на категория от данни като използва графични помощни средства • Синхронизира избора на ред от графичната таблица със стойностите на полетата, извеждани в други графични компоненти с графични помощни средства • Разработва програмна реализация на събитие за запазване на редактираните стойности в полетата от таблицата 	<ul style="list-style-type: none"> • Графична компонента за таблично извеждане на данни (DataGridView / JTable) • Свързването на графична компонента към категория от данни • Синхронизиране на графични компоненти с данни от база от данни
1.6 Графичен интерфейс от вида „едно-към-много“	<ul style="list-style-type: none"> • Генерира обектно ориентиран модел по таблици на зададена база от данни, свързани с релация „едно-към- много“, за моделиране на решение на примерни информационни проблеми • Генерира приложение с графичен интерфейс от вида „едно-към-много“ (Master-Detail/Parent-Child) на свързани таблици с релация „едно-към-много“ посредством графични помощни средства и обяснява генерирания изходен код • Дискутира приложения, където е подходящо да се използва графичен интерфейс от вида „едно-към- много“ • Подбира подходящи компоненти на графичния интерфейс 	<ul style="list-style-type: none"> • Релация „едно-към-много“ между класове (entity) на обектно ориентиран модел • Насоченост на релация между класове (entity) на обектно ориентиран модел • Графичен интерфейс от вида „едно-към- много“ (Master-Detail/ Parent-Child)
1.7 Въведение в интегриран език за дефиниране на заявки	<ul style="list-style-type: none"> • Интерпретира ключови думи на интегриран език за дефиниране на заявки при създаване на структура от данни, съдържаща записи в таблица на източник на данни • Прилага филтриране и сортиране на елементите на заявка посредством интегриран език за дефиниране на заявки • Коментира аналозите на интегриран език за дефиниране на заявки със стандартния език за изпълнение на структурни заявки в релационна база от данни • Създава приложение на обектно ориентиран модел на база от данни за извеждане записи от таблица с филтриране и сортиране 	<ul style="list-style-type: none"> • Параметър за тип на структура от данни на заявка • Дефиниране на заявка на интегриран език, филтриране и сортиране • Изпълнение на заявка на интегриран език с програмни средства, обект от клас за управление на категории от данни

1.8 Съхраняване на категория (entity) от данни	<ul style="list-style-type: none"> • Генерира обектно ориентиран модел по таблици на зададена база от данни за моделиране на решение на конкретен проблем • Създава графичен интерфейс за въвеждане на категория от данни като използва графични помощни средства. • Свързва графичните компоненти към съответните полета на категория от данни като използва графични помощни средства • Използва класа за управление на категориите от данни • за съхраняване на въведена категория (entity) от данни 	<ul style="list-style-type: none"> • Съхраняване на категория (entity) от данни • Метод за съхранение на категория данни
1.9 Редактиране на категория (entity) от данни	<ul style="list-style-type: none"> • Генерира обектно ориентиран модел по таблици на зададена база от данни за моделиране на решение на конкретен проблем • Създава графичен интерфейс за редактиране на категория от данни като използва графични помощни средства. • Свързва графичните компоненти към съответните полета на категория от данни като използва графични помощни средства • Използва класа за управление на категориите от данни за съхраняване на редактирана категория (entity) от данни 	<ul style="list-style-type: none"> • Редактиране на категория (entity) от данни • Метод за намиране на категория (entity) от данни за редактиране, потвърждение за редактиране
1.10 Изтриване на категория (entity)	<ul style="list-style-type: none"> • Генерира обектно ориентиран модел по таблици на зададена база от данни за моделиране на решение на конкретен проблем • Създава графичен интерфейс за изтриване на категория от данни като използва графични помощни средства. • Свързва графичните компоненти към съответните полета на категория от данни като използва графични помощни средства • Използва класа за управление на категориите от данни • за изтриване на избрана категория (entity) от данни 	<ul style="list-style-type: none"> • Изтриване на категория (entity) от данни • Метод за намиране на категория (entity) от данни за изтриване, потвърждение за изтриване
1.11 Филтриране и сортиране на данни от свързани категории (entities)	<ul style="list-style-type: none"> • Генерира обектно ориентиран модел по таблици на зададена база от данни, свързани с релация „едно-към- много”, за моделиране на решение на конкретен проблем • Създава графичен интерфейс към свързани категории от данни като използва графични помощни средства • Подбира и настройва свойствата на графичните компоненти към съответните категории данни • Прилага методи от класа за управление на категориите и интегриран език за дефиниране на заявки за филтриране и сортиране на данни от свързани категории (entities) при програмиране на събитията в графичния интерфейс 	<ul style="list-style-type: none"> •

1.12 Обработка на данни от свързани категории (entities)	<ul style="list-style-type: none"> • Генерира обектно ориентиран модел по таблици на зададена база от данни, свързани с релация „едно-към- много”, за моделиране на решение на примерни информационни проблеми • Създава графичен интерфейс към свързани категории от данни и резултати от обработка на данни като използва графични помощни средства. • Подбира и настройва свойствата на графичните компоненти към съответните категории данни • Прилага методи от класа за управление на категориите и интегриран език за дефиниране на заявки за обработка на данни (пресмятания) от свързани категории (entities) при програмиране на събитията в графичния интерфейс 	<ul style="list-style-type: none"> •
2. АНАЛИЗ, ПРОЕКТИРАНЕ И РЕАЛИЗИРАНЕ НА ПРИМЕРНИ ПРИЛОЖЕНИЯ		
2.1 Анализ на изискванията за информационна система	<ul style="list-style-type: none"> • Описва етапите в жизнения цикъл на информационна система • Анализира съответствието на изискванията за организация на информацията в зададен пример по отношение на зададен модел на релационни бази от данни и коригира допуснати пропуски в задания модел • Анализира изискванията за реализация на информационни процеси в конкретен пример • Обобщава, документира и представя изискванията 	<ul style="list-style-type: none"> • Жизнен цикъл на информационна система
2.2 Проектиране и програмиране на информационна система	<ul style="list-style-type: none"> • Работи в екип при моделиране на решение на част от анализирания проблем, създава обектно ориентиран модел на база от данни, проектира графичен потребителски интерфейс • Интерпретира модел за решаване на примерно зададен • информационен проблем Генерира програмен код за реализация на зададен • модел с помощта на стандартни библиотеки и графични помощни средства. 	<ul style="list-style-type: none"> • Архитектура на софтуерно приложение • Добър стил на графичен интерфейс на информационна система
2.3 Тестване и представяне на проект	<ul style="list-style-type: none"> • Описва работата от етапите на реализацията на софтуерен проект. • Създава тестови примери с входни данни и коментира получените резултати. • Представя проект пред аудитория. 	<ul style="list-style-type: none"> • Тестване, автоматизирани симулация на процес средства за тестване

Годишен брой часове за изучаване на модула в XII клас - 52 часа през втория учебен срок

Допълнителни уточнения за конкретния модул:

- Програмата предвижда обучението да се извършва на базата на език за визуално програмиране – Microsoft C#.NET) или Java по избор на преподавателя.
- Препоръчват се следните среди за програмиране: актуални версии на Microsoft Visual Studio (професионална или свободноразпространяема версия (за C#.NET), Netbeans или Eclipse (за Java)
- Препоръчва се обучението да се провежда предимно под формата на комбинирани уроци.
- Препоръчва се занятията да се провеждат в блок от два учебни часа и всеки ученик да разполага със собствено работно място за работа в съответната програмна среда.
- Препоръчва се описаните нови понятия да се разглеждат на концептуално ниво и да се използват при практическата работа.
- Часовете за преговор да включват предимно изпълнение на компютър на практически дейности по теми от учебната програма.
- Препоръчително е да се използват интегрираните среди NetBeans и Visual Studio .NET.
- За създаване на обектно ориентиран модел на бази от данни се препоръчва да се използва Java Persistence API (JPA) с Java Persistence Query Language (JPQL), EclipseLink Query Language (EQL) или Entity Framework (EF) с Language INtegrated Query (LINQ) посредством съответните помощни графични средства в интегрираните среди за програмиране.
- Да се идентифицират и прилагат добри практики на програмиране
- Препоръчва се по възможност на учениците да се предостави свободен достъп до компютърните зали извън редовните часове за упражнения и работа по проекти.

Препоръчително разпределение на часовете:

За нови знания	до 30 часа	до 50%
За преговор	до 15 часа	
Практически дейности	до 18 часа	до 50%
За контрол и оценка (За входно и изходно ниво, междинно контролно)	до 6 часа	

СПЕЦИФИЧНИ МЕТОДИ И ФОРМИ ЗА ОЦЕНЯВАНЕ НА ПОСТИЖЕНИЯТА НА УЧЕНИЦИТЕ

Съотношение при формиране на срочна и годишна оценка:

Текущи оценки от практически задания в клас	20%
Текущи оценки от домашни работи	20%
Оценки от работа по проект	10%
Оценка на изходно ниво	30%
Оценки от контролни работи	20%

ДЕЙНОСТИ И МЕЖДУПРЕДМЕТНИ ВРЪЗКИ

Дейности	Описание на дейност
<i>Дейност 1.</i>	Конфигуриране на източник на данни и създаване на връзка към реляционна база от данни с графични средства на интегрирана среда за програмиране. Учениците демонстрират познавателни и приложни компетентности при избор на доставчик на база от данни и параметрите, необходими за създаване на връзка към база от данни, както и при използване на помощни средства за стартиране, разглеждане на структурата на базата от данни и спирането на базата от данни.
<i>Дейност 2.</i>	Създаване на база от данни, таблица и въвеждане/редактиране на данни върху сървър от база от данни с помощни визуални средства на интегрирана среда за програмиране. Учениците демонстрират познавателни и приложни компетентности при дефиниране на свойствата на полета на таблица по предварително зададен модел на таблица като съобразяват типовете данни и останалите свойства на полетата на таблицата с изискванията на съответната реляционна база от данни, разграничават операции COMMIT и ROLLBACK с транзакции съответно при потвърждаване и отказване на въведени данни.
<i>Дейност 3.</i>	Архивиране и възстановяване на зададена база от данни, посредством скрипт на стандартен език (SQL) за структуриране на заявки. Учениците демонстрират умения за използване на помощни средства на интегрирана среда програмиране при архивиране и възстановяване на реляционна база от данни върху сървър за база от данни.
<i>Дейност 4.</i>	Генериране на обектно ориентиран модел на зададена база от данни с графични средства на средства на интегрирана среда за програмиране. Учениците демонстрират познавателни и приложни компетентности за структурата на обектно ориентиран модел и начина за генерирането му със средствата на интегрирана среда за програмиране. В комбинация с други от тук споменатите дейности тази дейност може да се използва за формулиране на проект.
<i>Дейност 5.</i>	Генериране на софтуерно приложение за извеждане на записите на таблица от база от данни в таблица на графичен прозорец като се използват визуални помощни средства. Учениците демонстрират познавателни и приложни компетентности по структурата на обектно ориентиран модел на бази от данни и приложението на визуални помощни средства за синхронизиране на таблична компонента от графичния интерфейс със записите на таблица от примерно зададена база от данни. В комбинация с други от тук споменатите дейности, тази дейност може да се използва за формулиране на проект
<i>Дейност 6.</i>	Генериране на софтуерно приложение с графичен интерфейс от вида „едно-към-много” (Master-Detail/Parent-Child) на свързани таблици с релация „едно-към-много” посредством графични помощни средства в интегрирана среда за програмиране. Учениците демонстрират познавателни и приложни компетентности как се интерпретира релация „едно-към- много ” в обектно ориентиран модел на така свързани таблици, предимствата на графичен интерфейс от вида „едно-към- много”, както и реализацията на този вид интерфейс с помощните визуални средствата на интегрирана среда за програмиране. В комбинация с други от тук споменатите дейности, тази дейност може да се използва за формулиране на проект.
<i>Дейност 7.</i>	Създаване на конзолно приложение на обектно ориентиран модел на бази от данни за извеждане записи от таблица с филтриране и сортиране. Учениците демонстрират познавателни и приложни компетентности при интерпретация на зададени команди на интегриран език за дефиниране на заявки по аналогия с командите на стандартен език за структуриране на заявки (SQL), както и при използване на средствата на обектно ориентиран език за изпълнение на тези команди - примерно команди за извеждане записи от таблица с филтриране и сортиране. В комбинация с други от тук споменатите дейности тази дейност може да се използва за формулиране на проект.

<i>Дейност 8.</i>	Създаване на интерактивен графичен интерфейс за въвеждане на категория от данни като използва графични помощни средства на интегрирана среда. Учениците демонстрират познавателни и приложни компетентности при подбор на графичните компоненти и помощните средства за синхронизацията им с базата от данни, а също и средствата на обектно ориентиран език за съхранение на категория от данни в съответната ѝ таблица на базата от данни. В комбинация с други от тук споменатите дейности, тази дейност може да се използва за формулиране на проект.
<i>Дейност 9.</i>	Създаване на интерактивен графичен интерфейс за редактиране на категория от данни като използва графични помощни средства на интегрирана среда. Учениците демонстрират познавателни и приложни компетентности при подбор на графичните компоненти и помощните средства за синхронизацията им с базата от данни, а също и средствата на обектно ориентиран език за редактиране на категория от данни и последващото ѝ съхранение в съответната ѝ таблица на базата от данни. В комбинация с други от тук споменатите дейности, тази дейност може да се използва за формулиране на проект
<i>Дейност 10.</i>	Създаване на интерактивен графичен интерфейс за изтриване на категория от данни като използва графични помощни средства на интегрирана среда. Учениците демонстрират познавателни и приложни компетентности при подбор на графичните компоненти и помощните средства за синхронизацията им с базата от данни, а също и средствата на обектно ориентиран език за изтриване на категория от данни от съответната ѝ таблица на базата от данни. В комбинация с други от тук споменатите дейности, тази дейност може да се използва за формулиране на проект.
<i>Дейност 11.</i>	Създаване на графичен интерфейс към свързани категории от обектно ориентиран модел на данни като използва визуални помощни средства. Учениците демонстрират познавателни и приложни компетентности да използват помощни средства на интегрирана среда, за да генерират релация „едно-към-много ” в обектно ориентиран модел на база от данни при решение на примерни информационни проблеми, както и да прилагат методи от класа за управление на категориите в обектно ориентиран модел на базата от данни и интегриран език за дефиниране на заявки за извличане и обработка на информация (филтриране, сортиране и изчисления) от свързани категории (entities) при програмиране на събитията в графичния интерфейс.
<i>Дейност 12.</i>	Създаване, документиране, тестване и представяне на проект по програмиране на информационни системи. Учениците демонстрират аналитични, познавателни и приложни компетентности при документиране на изискванията за изпълнение на проекта и етапите в жизнения му цикъл, генериране на програмен код за реализация на зададен модел с помощта на стандартни библиотеки и визуални помощни средства, тестване на полученото приложение, както и при структуриране и докладване на извършената работа.