По начальному тесту

|  |  |  |  |
| --- | --- | --- | --- |
| **Вопросы** | **К какому разделу относится** | **На какие темы вопрос опирается** |  |
| 1. Выберите верные утверждения относительно типов данных:  **- примитивный тип данных определяется языком программирования как базовая встроенная единица языка;**  - ссылочные переменные не могут принимать значение null;  - ссылочные типы данных включают в себя целочисленные (integer), дробные (floating-point) и булевы (boolean);  - простые переменные хранят ссылку на некоторый объект совместимого типа. | ООП. Типы данных | - объявление переменных;  - современное понятие типа в языках программирования;  - типы данных;  - примитивные и ссылочные типы данных; |  |
| 2. Выберите группу, относящуюся к примитивным типам данных:  - целочисленные (byte, short, int, long, float), дробные (float, double);  - целочисленные (byte, short, int, long, float), дробные (char, double), булевые (boolean);  **- целочисленные (byte, short, int, long, char), дробные (float, double), булевые (boolean);**  - целочисленные (byte, short, int), дробные (float, double, long), булевые (boolean). | ООП. Типы данных | - объявление переменных;  - современное понятие типа в языках программирования;  - типы данных;  - примитивные типы данных; |  |
| 3. Что относятся к объектным или ссылочным типам данных?  - Классы, интерфейсы.  - Массивы и перечисления.  - Булевые (boolean) типы данных.  **- Классы, интерфейсы и массивы.** | ООП. Типы данных | - объявление переменных;  - современное понятие типа в языках программирования;  - типы данных;  - ссылочные типы данных; |  |
| 4. Абстрактный метод – это…  - метод класса, реализация которого дана в развернутом виде;  **- член базового класса, который не предусматривает реализации по умолчанию, а предлагает только сигнатуру;**  - метод, который не имеет своей реализации в классе-наследнике, но он должен быть реализован в базовом классе;  - нет верного ответ. | ООП. Классы и интерфейсы | - классы и объекты;  - определение методов и свойств;  - абстрактные классы и методы |  |
| 5. Выберите верное утверждение о статических методах:  **- вызываются для целого класса, а не для каждого конкретного объекта, созданного на его основе;**  - они не могут выполняться, если не создан ни один экземпляр класса;  - они не могут непосредственно обращаться к другим статическим методам;  - статический метод может быть отменен, то есть стать не статическим. | ООП. Классы и интерфейсы | - классы и объекты;  - определение методов и свойств;  - статические члены класса |  |
| 6. Интерфейс – это:  - конструкция в коде программы, используемая для поддержки динамического выбора (resolution) переменных во время выполнения программы;  - метод, реализация для которого дана в развернутом виде;  **- набор констант и абстрактных методов;**  - нет верного ответа. | ООП. Классы и интерфейсы | - классы и объекты;  - абстрактные классы и методы;  - интерфейсы |  |
| 7. Инкапсуляция представляет собой:  **- механизм программирования, объединяющий в рамках класса переменные и методы и рассматривающий получившийся объект как единое целое;**  - способности языка расширять детали реализации пользователю объекта;  - распаковку данных и/или функций по компонентам. | ООП. Принципы | - принципы объектно-ориентированного программирования;  - инкапсуляция; |  |
| 8. Полиморфизм – это:  - способность классов решать похожие задачи исключительно одинаковыми способами;  **- способность классов решать похожие задачи разными способами;**  - свойство, позволяющее использовать один и тот же интерфейс для разных классов действий. | ООП. Принципы | - принципы объектно-ориентированного программирования;  - полиморфизм;  - перегруженные методы |  |
| 9. В чем заключается смысл принципа наследования?  **- Не надо каждый раз заново (с нуля) описывать новый объект, а можно указать родителя (базовый класс) и описать отличительные особенности нового класса.**  - Один объект может наследовать свойства другого объекта, но не может добавлять к ним черты, характерные только для него.  - Объект, принадлежащий классу-родителю, может использовать поля, свойства и методы класса-потомка.  - Все варианты ответов. | ООП. Принципы | - принципы объектно-ориентированного программирования;  - наследование;  - переопределенные методы |  |