**Java level 1 30 rows**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Для объявления переменной достаточно задать? | | Имя | Тип, имя, первоначальное значения | | | Тим, имя | | | 3 | | |
| Возможно ли в Java использовать выражение вида a < x < b? | | Да | Нет | | | Да, но не рекомендуется | | | 2 | | |
| Результат сравнения – это: | | Логическое значение | Объект | | | Символ | | | 1 | | |
| С каким расширением хранится исходный код классов, а с каким скомпилированный? | | Исходный код в .java, скомпилированный в .class | Исходный код в .java, скомпилированный в .javac | | | Исходный код в .javac, скомпилированный в .class | | | 1 | | |
| Если в сигнатуре метода указано void, то ключевое слово return в теле метода: | | Не обязательно, может быть указано с любым значением | Обязательно как минимум один раз с любым значением | | | Не обязательно, но может быть указано в качестве прерывания работы метода | | | 3 | | |
| Укажите правильную форму метода main | | public static void main (String[] args) | public void main (String[] args) | | | public static void Main (String[] args) | | | 1 | | |
| Что такое значение переменной? | | Участок данных, где хранятся данные | Данные, хранящиеся в переменной | | | Операция, доступная для действия с переменной | | | 2 | | |
| Что означает декларируемый Java принцип "Write once, run anywhere" ? | | Java-программа может быть скомпилирована в исполняемый файл под любую платформу. | Скомпилированная Java-программа может быть запущена на любой платформе, где есть JVM. | | | Java-программа может быть запущена из исходников на любой платформе, где есть интерпретатор Java. | | | 2 | | |
| Как называется двоичный формат, который понимает виртуальная машина Java? | | Virtual code (виртуальный код) | Intermediate language (промежуточный язык) | | | Bytecode (байткод) | | | 3 | | |
| Предположим, вы написали программу, состоящую из двух классов, и в каждом классе объявили точку входа — метод main. Что из этого выйдет? | | Программа не скомпилируется | Программа скомпилируется и запустится. Неоднозначности не возникнет, поскольку при запуске всегда явно указывается класс, в котором JVM должна искать метод main | | | Программа скомпилируется, но не запустится | | | 2 | | |
| Выберите верное утверждение про формат JAR. | | Формат JAR основан на формате ZIP. | В JAR-файл разрешено запаковывать только class-файлы. | | | Формат JAR основан на формате RAR. | | | 1 | | |
| Что такое JDK? | | Набор инструментов для разработки Java-программ, включая компилятор | Набор инструментов для запуска Java-программ, включая виртуальную машину | | | Виртуальная машина Java | | | 1 | | |
| Выберите из приведенных ниже комментариев те, которые являются JavaDoc-комментариями. | | /\*\* Is this a javadoc? \*/ | /// Is this a javadoc? | | | /\*! Is this a javadoc? \*/ | | | 1 | | |
| Укажите эквивалентные булевские операторы | | ^ и != | & и == | | | | и != | | | 1 | | |
| Укажите размер целочисленного типа int в Java. | | 32 бита | 16 бит | | | Размер динамически изменяется, чтобы вместить любое нужное пользователю целое число. | | | 1 | | |
| Зачем в Java, объектно-ориентированном языке, есть примитивные типы, не являющиеся объектами? Ведь есть полноценные объекты-аналоги java.lang.Integer и т.п. | | Примитивные типы — пережиток ранних версий Java. Теперь их не могут убрать из-за обратной совместимости. | Примитивные типы — это просто синонимы для соответсвующих ссылочных типов. Т.е. int — синоним для java.lang.Integer и т.п. | | | Примитивные типы нужны для производительности вычислений. Работа с объектами намного медленнее. | | | 3 | | |
| Для какого преобразования не требуется явно писать в программе оператор приведения типа? | | float -> long | String -> int | | | int -> long | | | 3 | | |
| Выберите выражение, которое даст в качестве результата строку A12. | | 'A' + "12" | 'A' + '1' + "2" | | | "A" + ('\t' + '\u0004') | | | 1 | | |
| Значения каких типов могут использоваться в операторе switch? | | float | String | | | boolean | | | 2 | | |
| В чем разница между классом и объектом? | | В Java есть только классы, объектов не бывает. | Объект — это конкретный экземпляр класса. | | | Разницы нет, это одно и то же. | | | 2 | | |
| JAVA  **Уровень 2** | | | | | | | | | | | |
| Что такое класс в Java? | | Базовый элемент объектно-ориентирован­ного программирования в языке Java. | Просто одно из возможных названий переменной. | | | Уровень сложности программы. Все операторы делятся на классы в зависимости от сложности их использования. | | | 1 | | |
| Как объявить класс в коде? | | class MyClass {} | new class MyClass {} | | | MyClass extends class {} | | | 1 | | |
| Для чего используется оператор NEW? | | Для создания новой переменной. | Для объявления нового класса. | | | Для создания экземпляра класса. | | | 3 | | |
| Что означает ключевое слово extends? | | Что это дополнительный модуль класса, который расширяет его свойства. | Что данный класс наследуется от другого. | | | Что два класса делают одно и то же. | | | 2 | | |
| Что означает перегрузка метода в Java (overload). | | Несколько методов с одинаковым названием, но разным набором параметров. | Изменение поведения метода класса относительно родительского. | | | Изменение поведения метода класса относительно дочернего. | | | 1 | | |
| Что означает переопределение метода в Java (override). | | Изменение поведения метода класса относительно дочернего. | Изменение поведения метода класса относительно родительского. | | | Несколько разных классов с одинаковым методом. | | | 2 | | |
| Чем отличаются static-метод класса от обычного метода класса. | | Обычный метод класса можно переопределить, а static-метод нельзя. | Обычный метод класса работает от объекта класса, а static-метод от всего класса. | | | Обычный метод класса можно перегрузить, а static-метод нельзя. | | | 2 | | |
| Как вызвать static-метод внутри обычного? | | Можно, надо перед этим переопределить обычный метод класса. | Никак, static-метод можно вызвать только от объекта класса. | | | Можно, ничего дополнительно делать не надо. | | | 3 | | |
| Как вызвать обычный метод класса внутри static-метода? | | Никак, static-метод не работает с объектом класса. | Можно, ничего дополнительно делать не надо. | | | Можно, надо перед этим переопределить обычный метод класса. | | | 1 | | |
| Для чего необходимо ключевое слово this | | Это не ключевое слово. | Это указатель на текущий объект класса внутри самого класса. Его можно опускать при вызове метода класса, но лучше этого не делать. | | | Это указатель на переопределенный метод класса. Его нельзя опускать при вызове, иначе переопределение не сработает. | | | 2 | | |
| Подклассом класса IOException является: | | FileNotFoundException | | Exception | | | RuntimeException | | | 1 |
| Подклассом класса RuntimeException является: | | IOException | | FileNotFoundException | | | NullPointerException | | | 3 |
| Как называется класс, объявленный внутри метода? | | внутренний | | вложенный | | | локальный | | | 3 |
| В чем особенность daemon-потока? | | не может быть остановлен | | завершит свою работу по завершению работы всех обычных потоков | | | может быть перезапущен сколько угодно раз | | | 2 |
| Выберите неверный способ создания объекта: | | HashSet<String> s = new TreeSet<String>() | | Set<String> s = new HashSet<String>() | | | HashSet<String> s = new LinkedHashSet<String>() | | | 1 |
| Параметр loadFactory у отображения HashMap по-умолчанию равен: | | 0.8 | | 1.0 | | | 0.75 | | | 3 |
| Какой из методов не может привести к приостановке потока? | | wait() | | join() | | | notify() | | | 3 |
| Объекты какого класса являются иммутабельными? | | StringBuffer | | String | | | StringBulder | | | 2 |
| LinkedHashSet хранит элементы в порядке | | случайном | | добавления | | | отсортированном через compareTo() | | | 2 |
| Назовите две группы исключений в Java: | | Exception и Error | | Handle и Unhandled | | | Checked и Unchecked | | | 3 |
| **Java**  **УРОВЕЬ 3** | | | | | | | | | | |
| Укажите суперкласс для классов Exception и Error? | Throwable | | | | Catched | | Errors | 1 | | |
| Выберите верное утверждение | HashMap допускает в качестве ключа использовать null, Hashtable - не позволяет | | | | и Hashtable, и HashMap не синхронизированы | | и Hashtable, и HashMap синхронизированы | 1 | | |
| Какой из вариантов создания объекта недопустим? | Collection<String, Integer> map = new HashMap<>(); | | | | Map<String, Integer> map = new HashMap<>(); | | NavigableMap<String, Integer> map = new TreeMap<>(); | 1 | | |
| Какой метод в HashMap предоставляет возможность получить ключ по значению? | findKey() | | | | getKey() | | get() | 2 | | |
| Если несколько потоков должны работать с отображением Map, какой из классов желательно использовать? | java.util.ParallelHashMap | | | | java.util.LinkedHashMap | | java.util.concurrent.ConcurrentHashMap | 3 | | |
| Какой метод нужно выполнить у класса Thread для запуска нового потока? | execute() | | | | run() | | start() | 3 | | |
| За что отвечает метод ensureCapacity(int n) в классе ArrayList? | возвращает true если емкость коллекции больше n | | | | увеличивает емкость коллекции до n, если она меньше n | | увеличивает емкость коллекции на n элементов | 2 | | |
| Выберите верное утверждение о ключевом слове final | нельзя объявить абстрактный класс с модификаторами final | | | | можно переопределить методы с модификатором final | | можно менять значение final поля | 1 | | |
| Выберите неверное утверждение | в HashSet элементы упорядочены по возрастанию | | | | LinkedHashSet хранит элементы в порядке добавления | | TreeSet представляет собой отсортированное множество | 1 | | |
| При попытке вызвать метод у не созданного объекта будет сгенерировано исключение: | IllegalArgumentException | | | | IOException | | NullPointerException | 3 | | |
| Какое условие необходимо выполнить для использования класса в качестве ключа HashMap? | переопределить методы hashCode() и equals() | | | | переопределить только метод equals() | | переопределить только метод hashCode() | 1 | | |
| Выберите верное утверждение | RuntimeException - checked, Exception - unchecked | | | | RuntimeException - unchecked, Exception - checked | | RuntimeException - checked, Exception - checked | 2 | | |
| При выполнении static synchronized метода в роли монитора выступает? | объект, у которого вызвали этот метод | | | | нельзя указывать модификатор synchronized у static метода | | класс | 3 | | |
| Начальная емкость HashMap по умолчанию равна | 12 | | | | 32 | | 16 | 3 | | |
| Выберите верное утверждение | Array и Vector не синхронизированы | | | | Array синхронизирован, Vector не синхронизирован | | Array не синхронизирован, Vector синхронизирован | 3 | | |
| Поля, объявленные в интерфейсах, по умолчанию имеют модификаторы | public static final | | | | public static | | private static final | 1 | | |
| Для создания класса Checked исключения необходимо | унаследовать его только от класса Exception | | | | унаследовать его от класса checked исключения | | унаследовать его от класса RuntimeException | 2 | | |
| В каком классе или интерфейсе определены методы wait(), notify() и notifyAll()? | Object | | | | Thread | | Runnable | 1 | | |
| Метод Thread.sleep() | приостанавливает работу программы | | | | приостанавливает работу потока, в котором был вызван без освобождения монитора | | приостанавливает выполнение потока main | 2 | | |
| Выберите неверное утверждение | интерфейс Map наследуется от интерфейса Collection | | | | интерфейс List не наследуется от интерфейса Collection | | интерфейс Set не наследуется от интерфейса Collection | 1 | | |

**Python level 1**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Что выведет программа?print(15//2) | 7.5 | 7 | 8 | 2 |
| Что обозначает логический оператор == | Меньше | Равно | Не равно | 2 |
| Какая библиотека отвечает за время? | time | Time | clock | 1 |
| Как получить данные от пользователя? | Использовать метод get() | Использовать метод read() | Использовать метод input() | 3 |
| Что будет показано в результате?  name = "John" print('Hi, %s' % name) | Ошибка | "Hi, John" | "Hi, name" | 2 |
| В какой версии python появился оператор "морж"(:=)? | Python 3.4 | Python 3.8 | Python 3.7 |  |
| Какие built-in methods существуют в последней версии python? | reduce(), defaultdict(), help(), oct() | div(), slice(), memoryview(), \_\_import\_\_() | any(). compile(), exec(), bytesarray() | 3 |
| Какие методы list никогда не меняют существующий список. Возможно, какие-то из перечисленных методов не существуют. | copy(), index(), count() | div(), slice(), memoryview(), \_\_import\_\_() | reduce(), defaultdict(), help(), oct() | 1 |
| import this вернет дзен питона Какого утверждения в дзене питона нет? | Beautiful is better than ugly | Все эти утверждения есть в дзене | Readability counts | 2 |
| Какого понятия в python не существует? | Python simplification | Name mangling | List comprehension | 1 |
| Выберите возможный способ  задания списка | d = list | r = lists | 1b = [1, 3] | 1 |
| Что будет напечатано при исполнении следующего кода?  .print(type(1 / 2)) | type 'int' | type 'double' | type 'number' | 1 |
| **Строковый метод chr() -** | Преобразует число в строку | Преобразует символ в целое число | Преобразует целое число в символ | 3 |
| Какой символ выведет программа: s = 'school' print(s[-4]) | o | h | c | 2 |
| Имя строки.find() : | Определяет длину строки | Ищет в строке заданную подстроку | Определяет, состоит ли строка только из цифр | 2 |
| Сколько библиотек можно импортировать в один проект? | Не более 5 | Не более 23 | Неограниченное количество | 3 |
| Какая функция выводит что-либо в консоль? | out(); | write(); | print(); | 3 |
| Как получить данные от пользователя? | Использовать метод cin() | Использовать метод input() | Использовать метод read() | 2 |
| Где переменная правильно создана? | num = float(2) | int num = 2 | var num = 2 | 1 |
| Переменная int: | целая переменная | вещественная переменная | символьная строка | 1 |

**Python level 2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Выберите верное утверждение про списки: | Списки могут содержать элементы различных типов | Списки неизменяемые | Проверка на вхождение элемента в список происходит за константное время | 1 |
| Выберите верное утверждение про словари: | Поиск ключа в словаре происходит за линейное время | Словари изменяемые | Словари неизменяемые | 2 |
| К чему приведет обращение к непустому списку по индексу -1? | Вернется последний элемент | Будет ошибка IndexError | ошибка KeyError | 3 |
| Какая разница между sort() и sorted()? | Sort сортирует исходный список, а sorted возвращает новый | Функции sorted не существует | Sorted сортирует исходный список, а sort возвращает новый | 1 |
| Что верно про метод   set\_a.difference(set\_b)? | Возвращает True, если два множества одинаковые, False если хотя бы один элемент не совпадает | Возвращает множество из элементов, которые встречаются только в множестве set | Возвращает True, если два множества разные, False если хотя бы один элемент совпадает | 2 |
| В каком варианте правильно объявляется множество? | a = {} | a = [] | a = () | 1 |
| Что такое frozenset? | Множество, которое используется для хранения констант | Множество, которое хранит в себе только неизменяемые объекты | Множество, которое нельзя изменять | 3 |
| Чем отличаются методы remove() и discard(), применяемые к множеству? | Метода discard() для множеств не существует | Remove() удаляет элемент если он есть, но бросает ошибку если элемента нет, discard() просто удаляет элемент если он есть | discard() удаляет элемент если он есть, но бросает ошибку если элемента нет, remove() просто удаляет элемент если он есть | 2 |
| Элемент под каким индексом будет выдан из множества при использовании метода pop()? | Неизвестно, будет выдан случайный элемент | Никакой, потому что множество – неупорядоченная последовательность | -1, то есть последний | 1 |
| Какой метод используется для добавления элемента в множество? | append() | new() | add() | 3 |
| Что такое множество в Python? | Это список, содержащий вложенные списки в себе | Это контейнер, значения в котором не повторяются | Это список, содержащий в себе только функции | 2 |
| Что такое функция? | Объект, принимающий аргументы и возвращающий значение | Код, заключенный в круглые скобки | Структура, определяющая поведение объекта | 1 |
| Какое ключевое слово используется для создания функции? | def | function | fun | 1 |
| Для чего нужно ключевое слово global? | Чтобы сделать переменную доступной из любой точки кода | Чтобы переменную можно было изменять за пределами текущей области видимости | Такого ключевого слова в Python нет | 2 |
| Сколько параметров может принимать функция? | Бесконечно много | 100 | Нисколько, функция не принимает значения, только возвращает | 1 |
| Как расшифровывается LEGB? | Local, Enclosed, Global, Built-in | Light Edged Great Button | Local Environment of Global Bench | 1 |
| Python 3, в отличии от Python 2, полностью поддерживает Unicode. | Не верно, так как без ограничений Unicode поддерживают обе версии | Верно | Не верно, так как Unicode нормально не поддерживают обе версии | 2 |
| Какие из перечисленных выражений не создадут список из трех элементов: | list(range(3)) | 'a b c'.split(' ') | 'asd'.split() | 3 |
| Какой параметр обязательно принимает в себя метод экземпляра класса? | Тип объекта | Название класса | Сам экземпляр объекта | 3 |
| Как много работающих конструкторов в классе может иметь Python? | 2 и более | 1, в Python можно создать лишь один конструктор | 0, в них нет необходимости | 2 |

Python level 3 20 rows

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Функция определенная внутри класса называется - ... | Функция-класс | Метод | Операция | 2 |
| Как создать конструктор класса А? | def \_\_init\_\_(параметры конструктора) | def \_\_A\_\_(параметры конструктора) | def init(параметры конструктора) | 1 |
| Что из следующего верно о кортежах? | Кортеж занимает больше памяти чем список | Кортежи можно складывать | У кортежа есть метод append() | 1 |
| Что такое кортеж? | Упорядоченная изменяемая коллекция объектов одного типа | Неупорядоченная неизменяемая коллекция объектов одного типа | Упорядоченная неизменяемая коллекция объектов произвольных типов | 3 |
| Как сделать поле модели необязательным? | Интерфейс администратора рассматривает все поля как необязательные | Добавить к полю blank=True | Интерфейс администратора рассматривает все поля как необязательные | 2 |
| Какая функция принимает имя шаблона и контекст и возвращает объект HttpResponse с обработанным шаблоном? | render\_template | render\_and\_load | render\_to\_response | 3 |
| Какая команда manage.py создает таблицы базы данных, руководствуясь описанными моделями? | syncdb | validate | createtables | 1 |
| Какой атрибут формы используется для вывода конкретной метки поля? | label | print\_name | verbose\_name | 1 |
| Чем полезны лямбда-функции? | Они могут быть полезны как быстрые, однострочные функции. | Лямбда-функции используются для функционального программирования. | Лямбда-функции всегда упрощают чтение кода. | 1 |
| Какой метод функционального программирования используют лямбды? | filter() | reduce() | map() | 1 |
| Каким будет результат следующего фрагмента кода? (lambda x: (x + 3) \* 5 / 2)(3) | SyntaxError | 15.0 | 30.0 | 2 |
| Что делает threading.Lock? | Синхронизирует потоки | Ждет, пока поток не будет завершен | Передает сообщения между потоками | 1 |
| Сколько процессоров (или ядер) может использовать threading библиотека Python одновременно? | Два | Все доступные процессоры | Один | 3 |
| Что делает Thread.join()метод? | Добавляет поток в пул | Объединяет два потока в один | Дожидается завершения потока | 3 |
| Почему использовать a ThreadPoolExecutor в качестве диспетчера контекста - это хорошая идея ? | Он передает исключения обратно в основную программу | Выглядит лучше | Он управляет запуском всех потоков и ожидает их завершения. | 3 |
| В Python строки… | Массив символов | str объекты | Неизменяемые | 1 |
| Что из следующего разделит строку input\_string при первых двух встречах буквы «е»? | 'e'.split(input\_string, 2) | 'e'.split(input\_string, maxsplit=2) | input\_string.split('e', maxsplit=2) | 1 |
| Строки Python имеют свойство, называемое «неизменяемость». Что это значит? | Вы можете обновить строку в Python с помощью конкатенации | Строки нельзя делить числами | Строки в Python нельзя изменить | 3 |
| Если вы хотите преобразовать список строк input\_listв одну строку с запятой между каждым элементом, что из следующего вы бы дали в качестве входных данных join()? | input\_list | string | ',' | 1 |
| Для данного файла dog\_breeds.txt, что из следующего является правильным способом открыть файл для чтения как текстовый файл? | open('dog\_breeds.txt', 'w') | open('dog\_breeds.txt', 'rb') | open('dog\_breeds.txt', 'r') | 3 |