1. Названия переменных.

начинаться с буквы или **$ или \_** или с 0-9.

После все что угодно.

Нельзя применять зарезервированные слова

Public private protected int double char yeld Boolean float byte switch while for abstract break continue case do finally try catch final new const goto import package static void this throw volatile transient super stricfp assert enum native interface synchronized else extends

1. Статиические блоки

Выполняются при первой загрузке класса. Могут проинициализировать переменные для анонимных класов. С их помощью инициализирут static поля.

1. Переопределение static метода.

И можно и не не можно будет использоваться тот кто первый был создан

1. Assert?

Нужен для генерации AssertEror при проверке правильности результата.

1. String StringBuilder StringBuffer

SB-быстрее с точки зрения соединения строк но дороже в памяти так как для этого делается новая строка а не как String просто имеет масив байт builder синхронизированый. Если подстрока исползуем String.

1. IO

Writer reader для 2 байт inputstrem out для потоков ввода вывода классы реализуют один метод read write эти класы абстрактны. InputStream s= new FileInputStream(new File(“hela.txt”));

1. Heap stack in Java?

Куча – динамически распределяемая облсть памяти создана пристарте jvm.

Создание объекта происходит в куче. Любой в этой куче глобален выден во всем приложении.

Стек lifо – память для вызова методов как только закончил вышел из стакка. Туда же попадают и переменные. Если в методе объект он в кучу но ссылка в стеке.

1. Разность heap stack

Стек одним потоком программы. Куча живет все время программы а стек локален. -Xms -Xmx для JVM чтобы определить макс обьем занимаемый в памяти. StackOwerflow outofmemory.Стек намного меньше кучи.

1. Память в java.

Стара и новая куча. Новая из eden первая(созданые из проги объекты. Иного из них живут не долго временники их собирает ГК.Когда она заполняется ГК выполняет быструю сборку мусора.minor colecion перемещая выживших в другую ) and survival выжившая (пережили ГК время от времени они перемещаются в Tuned – крупные синглтоны высокоуровневые менеджеры ресурсов если заполнено major collection) обрабатывает все места. Есть еще permanet Generation – мета данные jvm.

1. Garbage collector

Есть несколько алгоритмов. Все зависит от обстоятельств кого удалять например если очень много нуулевых объектов он их помечает кто со ссылкой в выживших, а первой куче все удаляет. Если много рабочих объектов в первой он помечает мусор и его удаляет остальное компонует. Бывет сам вызывается время от времени.

1. Инициализация переменных

Инит не статических переменных происходит после вызова конструктора по умолчанию по тому если обращаться к полю то оно может быть кидать NullpointExeption.

1. HashMap for Junior

В Хеш карту значения попадают вычисляя хешкод елемента который может быть переопределен у наследника. И по этому ключу ложится значение в один из индексов масива. Если изменить ссылку значения то поменяется и ключ это значит что мы получим другой елемент с этим же ключем. Если хеши будут рандомны то мы не сможем оттуда ничего достать только положить. Если однин тотже хеш то map превратиться в список в одной ячейки масива.

Maven.

Lifecycle:

1. Validate – проверка на правильность meta-inf
2. Compile-
3. Test- из предидущего шага
4. Package- все в jar or war
5. Integration-tast – в среду тестировния
6. Verify – коректность пакета и его качество
7. Install загоняет пакет для использования его в других зависимостях
8. Deploy -отправляет на сервер в репозиторий откуда прогромисты могут его использовать.

Генерится из pom.xml который содержит блок dependensies в нем описаны библиотеки которые нужны проекту. На validate проверяет а скачаны ли они. Также надо указывать версию библ.