**Private конструктор**

1. **Как создать объект в Java, если конструктор имеет модификатор private?**Пример:

**public** **class** Example {   
    **private** Example() { System.out.println("Hello World"); }   
    **public** **static** **void** main(String[] args) {   
        // ...   
} }

Ответ

Если не брать в расчет reflection, то объект можно создать только через статический метод, который должен вернуть экземпляр данного класса:

**public** **class** Example {   
    **private** Example() { System.out.println("Hello World");}   
    **static** Example create() { **return** **new** Example();}   
   
    **public** **static** **void** main(String[] args) {   
        Example.create();   
} }

Cоздание объекта очень напоминает паттерн Singleton (одиночка)

**public** **class** Example {   
    **private** **static** Example example = **new** Example();    
   
    **private** Example() { System.out.println("Hello World");}   
    **static** Example create() { **return** example;}   
   
    **public** **static** **void** main(String[] args) {   
        Example.create();   
} }

Разница состоит в том, что во втором случае, объект будет существовать в единственном экземпляре.

1. **Зачем используется private конструктор?**

- Его рекомендуют использовать в утильных классах. Ну в тех, которые состоят только из утильных статических методов. Это сделано для того, чтобы нельзя было создать ненужный инстанс класса. Еще, такой класс желательно сделать final, чтобы от него не могли отнаследоваться... Защита от идиотов... Ну и еще, если это вложеный паблик класс с приватным конструктором, чтобы можно было инстансы делать только внутри класса, где находится этот вложеный класс, а использовать этот инстанс можно бы было снаружи. Это очень удобно, если пишется какая-то объектная обертка над структурами данных..

- Я вот всегда стараюсь делать приватным все по максимуму, чем меньше паблик АПИ, **тем меньше надо на него бумажек писать**, имхо, золотое правило