##### **Задание 3. Ссылки на коллекции**

Определена иерархия классов

**class** MedicalStaff{}

**class** Doctor **extends** MedicalStaff{}

**class** Nurse **extends** MedicalStaff{}

**class** HeadDoctor **extends** Doctor{}

Укажите корректные и некорректные операторы. Дайте ответу пояснение.

|  |  |  |
| --- | --- | --- |
|  | correct | not correct |
| Doctor doctor1 = **new** Doctor(); | V |  |
| Doctor doctor2 = **new** MedicalStaff();  - экземпляр предка не может иметь тип потомка |  | V |
| Doctor doctor3 = **new** HeadDoctor(); | V |  |
| Object object1 = **new** HeadDoctor(); | V |  |
| HeadDoctor doctor5 = **new** Object();  - экземпляр предка не может иметь тип потомка |  | V |
| Doctor doctor6 = **new** Nurse();  - объект другого типа, хоть и с общим предком |  | V |
| Nurse nurse = **new** Doctor();  - объект другого типа, хоть и с общим предком |  | V |
| Object object2 = **new** Nurse(); | V |  |
|  |  |  |
| List<Doctor> list1= **new** ArrayList<Doctor>(); | V |  |
| List<MedicalStaff> list2 = **new** ArrayList<Doctor>();  - дженерики должны быть идентичны |  | V |
| List<Doctor> list3 = **new** ArrayList<MedicalStaff>();  - дженерики должны быть идентичны |  | V |
| List<Object> list4 = **new** ArrayList<Doctor>();  - дженерики должны быть идентичны |  | V |
| List<Object> list5 = **new** ArrayList<Object>(); | V |  |

##### **Задание 4. Применение коллекций**

Заполните таблицу.

|  |  |  |
| --- | --- | --- |
|  | Основная функциональность | Примеры типичного использования |
| Set | Для хранения набора элементов, когда элементы должны быть уникальны. Реализации могут поддерживать упорядоченность, сортировку. | Хранение любых множеств, например, вариантов выбора, всех возможных ошибок, типовых фраз, названий стран, дней недели (лучше в виде Enum). |
| List | Для хранения списков, позволяет обращаться к элементу по индексу. В основе имеет массив, но в отличии от массива размер List’а можно менять. | Нумерованные наборы элементов, такие как участники мероприятия, полученные программой пакеты данных. |
| Queue | Очередь позволяет хранить элементы для последующей их обработки в определенном порядке.  Поддерживает разную реализацию методов вставки, удаления и проверки элементов. | Очередь пакетов данных, буферизация. |
| Map | Содержит пары «ключ» - «значение», ключи уникальны. | properties файлы, хранение пар «работник» - «характеристики» |