##### **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(); |  |  |
| Doctor doctor2 = **new** MedicalStaff(); |  |  |
| Doctor doctor3 = **new** HeadDoctor(); |  |  |
| Object object1 = **new** HeadDoctor(); |  |  |
| HeadDoctor doctor5 = **new** Object(); |  |  |
| Doctor doctor6 = **new** Nurse(); |  |  |
| Nurse nurse = **new** Doctor(); |  |  |
| Object object2 = **new** Nurse(); |  |  |
|  |  |  |
| List<Doctor> list1= **new** ArrayList<Doctor>(); |  |  |
| List<MedicalStaff> list2 = **new** ArrayList<Doctor>(); |  |  |
| List<Doctor> list3 = **new** ArrayList<MedicalStaff>(); |  |  |
| List<Object> list4 = **new** ArrayList<Doctor>(); |  |  |
| List<Object> list5 = **new** ArrayList<Object>(); |  |  |

##### **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. |
| List | Для хранения списков, позволяет обращаться к элементу по индексу. | Номерованные наборы элементов, такие как |
| Queue |  |  |
| Map |  |  |