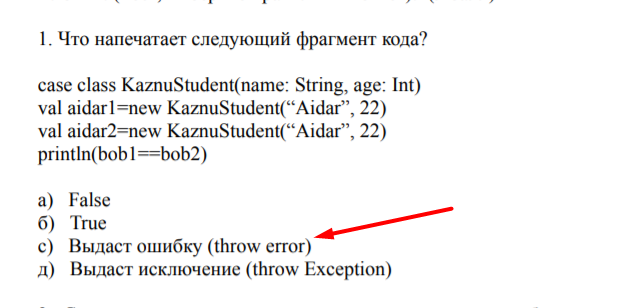
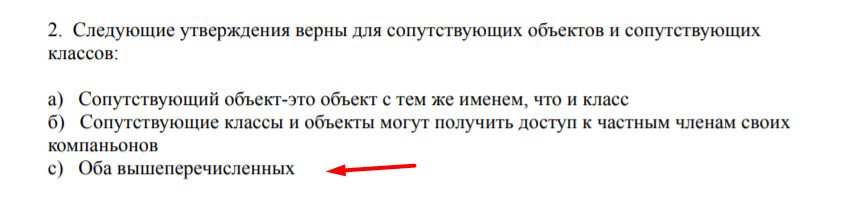
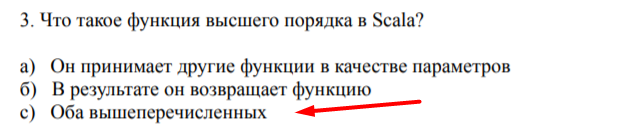
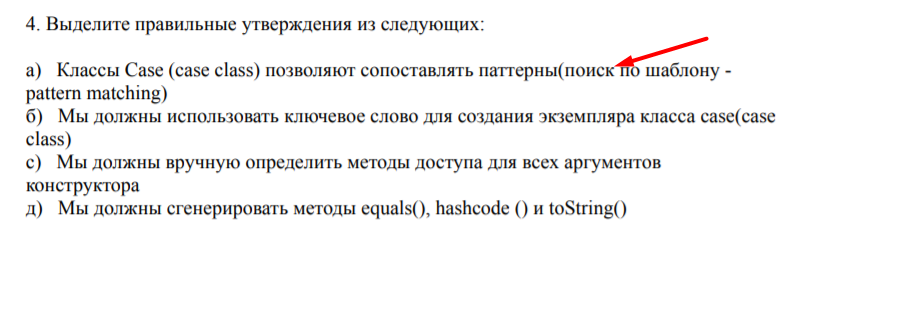
Блок А. (Тест, выберите правильный ответ): (9 балл)

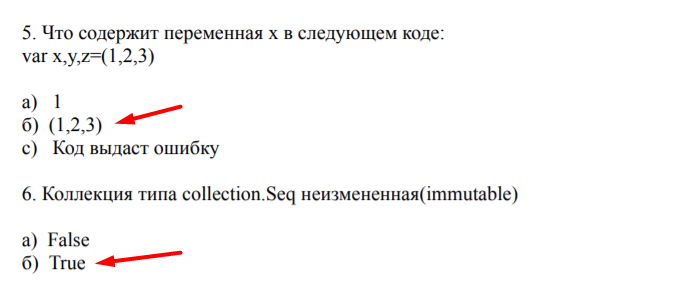


Нет переменных bob1 ,bob2. Если мы заменим их на aidar1 aidar2, мы получим True









**Блок Б. (Открытый вопрос) (12 балл)**

**Вопрос 1. Перечислите разницу между объектом и классом ?**

**Сlass** -это определение, описание. Он определяет тип с точки зрения методов и состава других типов.

**Оbject** - это singleton-экземпляр класса, который гарантированно уникален. Для каждого object в коде создается анонимный класс, который наследуется от любых классов, для реализации которых вы объявили object . Этот класс нельзя увидеть из исходного кода Scala , хотя вы можете добраться до него через отражение. Существует связь между object и class . Объект называется сопутствующим объектом класса, если они имеют одно и то же имя. Когда это происходит, каждый имеет доступ к методам видимости private в другом. Однако эти методы не импортируются автоматически. Вы должны либо импортировать их явно, либо префиксировать их именем класса/объекта.

**Вопрос 2. Что такое “Trait” в языке Scala, перечислите особенности ?**

Трейты (Traits) используются чтоб обмениваться между классами информацией о структуре и полях. Они похожи на интерфейсы из Java 8. Классы и объекты могут расширять трейты, но трейты не могут быть созданы и поэтому не имеют параметров.

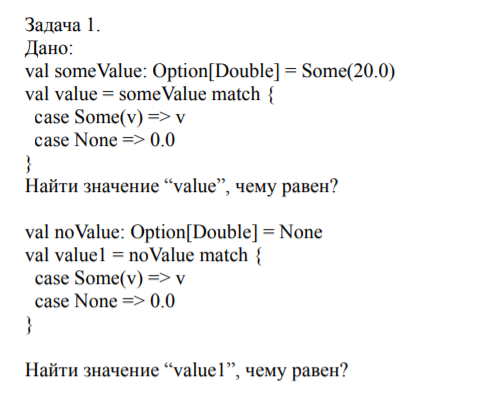
Минимальное объявление трейта - это просто ключевое слово trait и его имя: *trait Student*

Трейты наиболее полезны в качестве обобщенного типа с абстрактными методами. Чтоб использовать трейты, необходимо наследовать класс от него используя ключевое слово extends. Затем необходимо реализовать все абстрактные члены трейта, используя ключевое слово override. Туда где требуется определенный тип трейта, мы можем передавать любой наследованный от требуемого трейта класс.

**Вопрос 3. Что такое Case Class, перечислите особенности?**

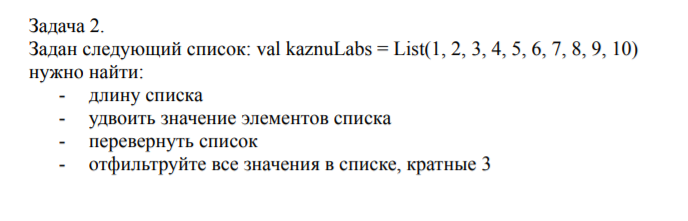
Классы образцы (Case classes) похожи на обычные классы с несколькими ключевыми отличиями, о которых мы поговорим ниже. Классы образцы хороши для моделирования неизменяемых данных. Минимальный вариант объявления класса образца: указание ключевого слова case class, название и список параметров (которые могут быть пустыми).

**Блок В. (Задача, требуется приложить ответ компилятора помимо решения самой задачи) (10 балл)**

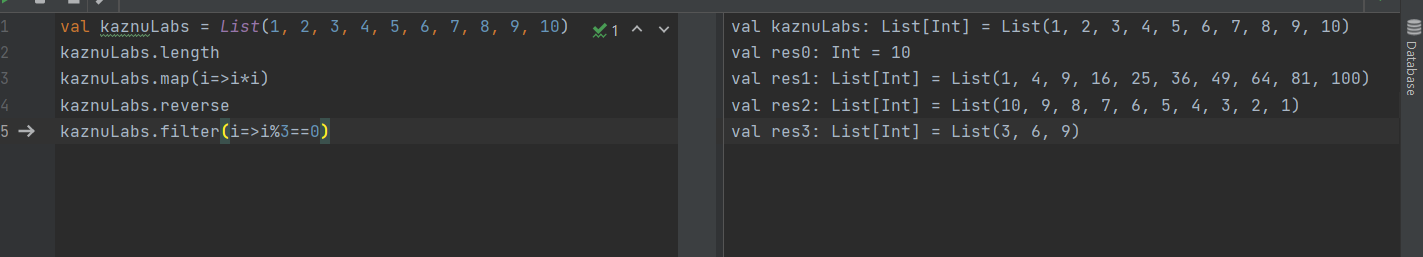


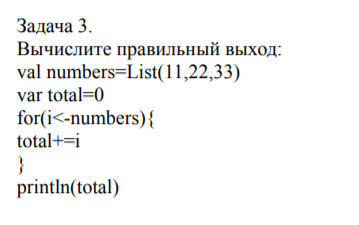
**Value Равен 20.0**

**Value1 Равен 0.0**

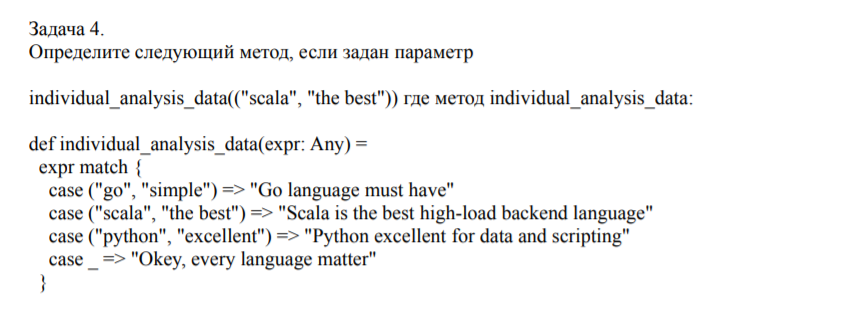


val kaznuLabs = *List*(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)  
kaznuLabs.length  
kaznuLabs.map(i=>i\*i)  
kaznuLabs.reverse  
kaznuLabs.filter(i=>i%3==0)





**Total=66**



Answer : **Scala is the best high-load backend language**

Блок Д. (Гитхаб загрузка ответов) (4 балл)