Q1.

|  |
| --- |
| Rice |
| price: double  code:String |
| Rice()  Rice(String code, double price)  Getter and setter |

|  |  |
| --- | --- |
| TC 1 | TC 2 |
| Enter Rice code: vn12  Enter Rice price: 400  1. TC = 1 - test getCode()  2. TC = 2 - test setPrice()  Enter TC: 1  OUTPUT:  VN12 | Enter Rice code: vn12  Enter Rice price: 400  1. TC = 1 - test getCode()  2. TC = 2 - test setPrice()  Enter TC: 2  Enter new Rice price: 560  OUTPUT:  560.00 |

Q2.

|  |
| --- |
| Rice |
| price: double  code:String |
| Rice()  Rice(String code, double price)  Getter and setter  toString() : code price |

|  |
| --- |
| BlackRice |
| maker: String |
| BlackRice()  BlackRice(String code, String maker,double price)  Getter and setter  double getPrice(): Nếu code mà bắt đầu với B hoặc b thì giá tăng 10% còn không thì bằng giá ban đầu  toString(): code maker price |

|  |  |
| --- | --- |
| TC 1 | TC 2 |
| Enter Rice code: vn12  Enter Rice price: 400  Enter Rice maker: red  1. TC = 1 - test toString()  2. TC = 2 - test getPrice()  Enter TC: 1  OUTPUT:  vn12 400.0  vn12 red 400.0 | Enter Rice code: b12  Enter Rice price: 400  Enter Rice maker: black  1. TC = 1 - test toString()  2. TC = 2 - test getPrice()  Enter TC: 2  OUTPUT:  440.00 |

Q3.

|  |
| --- |
| Rice |
| price: double  code:String |
| Rice()  Rice(String code, double price)  Getter and setter |

**The interface IRice is given (DO NOT CREATE THIS ONE).** Design and code a class named **MyRice** which will implement interface IRice and complete 3 methods which were declared in IRice:

|  |
| --- |
| ***<<interface>>***  ***IRice*** |
|  |
| +f1(List<Rice> a):void  +f2(List<Rice> a, double d):int  +f3(List<Rice> a):void |

|  |  |  |  |
| --- | --- | --- | --- |
| |  | | --- | | **MyRice** | |  | |  | |

* void f1(List<Rice> a) – Nếu giá của các Rice lớn hơn bang giá của Rice cuối cùng trong danh sách a thì tăng 10%.
* int f2(List<Rice> a, double d) – trả về số phần tử Rice mà có giá >= giá được đưa ra (d)*.*
* void f3(List<Rice> a) – xóa đi những Rice có giá lớn hơn giá của Rice cuối trong danh sách a.

|  |  |
| --- | --- |
| TC 1 | Add more how many Rice: 1  Rice code: vn30  Rice price: 58  Enter test function (1-f1;2-f2;3-f3): 1  OUTPUT:  FS21 66.00  KS20 74.80  FF12 52.00  vn30 63.80 |
| TC 2 | Add more how many Rice: 1  Rice code: vn30  Rice price: 58  Enter test function (1-f1;2-f2;3-f3): 2  Enter given Rice price: 60  OUTPUT:  2 |
| TC 3 | Add more how many Rice: 1  Rice code: vn30  Rice price: 58  Enter test function (1-f1;2-f2;3-f3): 3  OUTPUT:    FF12 52.00  vn30 58.00 |

Q4.

You are given an interface named **IString (DO NOT CREATE THIS ONE)**. Design a class **MyString** which will implement the interface Processible.

|  |  |  |  |
| --- | --- | --- | --- |
| |  | | --- | | ***<<interface>>***  **IString** | |  | | +sum(String string):int  +getCode(String string):String | |
|  |
| |  | | --- | | **MyString** | |  | |  | |

1. int sum(String string): nhập vào 1 dãy xâu số (các số cách nhau bởi dấu cách, tính tổng các số lớn hơn số cuối cùng (nếu có chữ thì tổng = 0)

2. String getCode(String string): nhập vào 1 xâu với các phần tử là số hoặc chữ, trả về xâu chỉ gồm toàn bộ các phần tử xâu (không có số) nhưng đảo ngược (phần tử cuối lên đầu, phần tử sát cuối thành thứ 2…. Phần tử đầu =0)

The program output might look something like:

|  |  |
| --- | --- |
| TC 1  Correct | 1. TC = 1 - test sum()  2. TC = 2 - test getCode()  Enter TC: 1  Enter a value in a string: 12 45 23 67 30  OUTPUT:  112 |
| TC 1  Not correct | 1. TC = 1 - test sum()  2. TC = 2 - test getCode()  Enter TC: 1  Enter a value in a string: 23 fg 45 mm  OUTPUT:  0 |
| TC 2 | 1. TC = 1 - test sum()  2. TC = 2 - test getCode()  Enter TC: 2  Enter a value of rc: ab 34 dc 45 hh  OUTPUT:  hh dc ab |