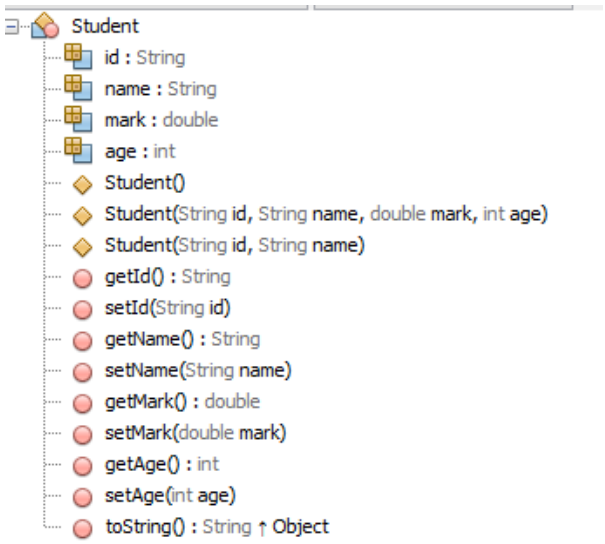
PRO192 LAB 6

Create a class named Student with some attributes and methods as shown below



- Suppose the attribute 'name' has at least two words, method **getName()** returns a string by setting the first character of each word in the name to uppercase, the rest to lowercase.
For example: name: **nguyen son tung**
getname will return: **Nguyen Son Tung**
  - toString return a string in format: id, name, age, mark

There is a ready-made **IStudent** interface you can use without creating **IStudent.java**

Create a class named **StudentList** to implement the **IStudent** interface then complete all its abstract methods.
The functionality of all methods is described in detail below.

```
 /**
    * for f1: There is a file containing student information in the form:
                id; name; age; mark
   * Each student in separate lines The f1_readfromfile()
    * method reads all students in the file into a list t;
    */
   public void f1_readfromfile(RandomAccessFile f, List<Student> t);

   /**
    * for f2: Write a list of student to the file in the form: id;    name;  age;     mark
Each field separated by a semicolon + a tab and each student is in separate lines
    */
   public void f2_writetoFile(List<Student> t, RandomAccessFile f);

   /**
    * for f3: The attribute 'name' is the full name so there can be several
    * words, student name is the last word. Sort the list of students by name
    * in ascending order. Write the results to the file f
    */
   public void f3_sortbyName(List<Student> t, RandomAccessFile f);

   /**
    * for f4: Sort the list of students by mark in descending order. Write the
```

```
     * results to the file f
     */
    public void f4_sortbyMark(List<Student> t, RandomAccessFile f);

    /**
     * for f5: Get from a list of students whose surname is equal “Nguyen”. Write the
     * results to the file f
     */
    public void f5_getSurName(List<Student> t, RandomAccessFile f);

    /**
     * for f6: Select the top six students from the list of high-scoring  (Updated)
     * students. Write the result to file f. If more than six students have the same high-scoring then
get all of them.
     */
    public void f6_getTop5(List<Student> t, RandomAccessFile f);

    /**
     * for f7: Taken from the list of students with the highest scores. Write
     * the result to file f
     */
    public void f7_getmax(RandomAccessFile f, List<Student> t);

    /**
     * for f8: Taken from the list of students with the lowest scores. Write the
     * result to file f
     *
     */
    public void f8_getmin(RandomAccessFile f, List<Student> t);

    /**
     * for f9: Write a list of student to the file in the form: id;    name;  age;     mark;   status
In which if mark>5 status is passed else status is Not passed  (Updated)
Each student is in separate lines.
     */
    public void f9_getFull(RandomAccessFile f, List<Student> t);

    /**
     * for f10
     *
     * Taken from the list of students under 20 years old. Write the result to file f
     */
    public void f10_getunder20(RandomAccessFile f, List<Student> t);
```

Note:

Please submit the entire project in a .zip or .rar file with the file name as your student ID (StudentID-Student_Name). **DO NOT** use Vietnamese characters to name your project files.

For example: **He170123 - Nguyen Huyen Trang.rar**

```
Enter TC (1-10): 1

He190001;       Vu Van Huy ;    18;     9.5
He190005;       Pham Thanh Nam ;        22;     7.0
He190006;       Bui Van Tan ;   18;     8.7
He190010;       Vu Minh Tuan ;  18;     8.0

Expected:
He190001;       Vu Van Huy ;    18;     9.5
He190005;       Pham Thanh Nam ;        22;     7.0
He190006;       Bui Van Tan ;   18;     8.7
He190010;       Vu Minh Tuan ;  18;     8.0

Test case 1 is OK
BUILD SUCCESSFUL (total time: 3 seconds)
```

```
Enter TC (1-10): 2
OUTPUT:
He190001;       Vu Van Huy ;    18;     9.5
He190005;       Pham Thanh Nam ;        22;     7.0
He190006;       Bui Van Tan ;   18;     8.7
He190011;       Nguyen Hong An ;        18;     5.5
He190010;       Vu Minh Tuan ;  18;     8.0

Expected:
He190001;       Vu Van Huy ;    18;     9.5
He190005;       Pham Thanh Nam ;        22;     7.0
He190006;       Bui Van Tan ;   18;     8.7
He190011;       Nguyen Hong An ;        18;     5.5
He190010;       Vu Minh Tuan ;  18;     8.0

Test case 2 is OK
BUILD SUCCESSFUL (total time: 8 seconds)
```

```
Enter TC (1-10): 3
OUTPUT:
He190011;       Nguyen Hong An ;        18;     5.5
He190001;       Vu Van Huy ;    18;     9.5
He190005;       Pham Thanh Nam ;        22;     7.0
He190006;       Bui Van Tan ;   18;     8.7
He190010;       Vu Minh Tuan ;  18;     8.0

Expected:
He190011;       Nguyen Hong An ;        18;     5.5
He190001;       Vu Van Huy ;    18;     9.5
He190005;       Pham Thanh Nam ;        22;     7.0
He190006;       Bui Van Tan ;   18;     8.7
He190010;       Vu Minh Tuan ;  18;     8.0

Test case 3 is OK
BUILD SUCCESSFUL (total time: 2 seconds)
```

```
Enter TC (1-10): 4
OUTPUT:
He190001;       Vu Van Huy ;    18;     9.5
He190006;       Bui Van Tan ;   18;     8.7
He190010;       Vu Minh Tuan ;  18;     8.0
He190005;       Pham Thanh Nam ;        22;     7.0
He190011;       Nguyen Hong An ;        18;     5.5

Expected:
He190001;       Vu Van Huy ;    18;     9.5
He190006;       Bui Van Tan ;   18;     8.7
He190010;       Vu Minh Tuan ;  18;     8.0
He190005;       Pham Thanh Nam ;        22;     7.0
He190011;       Nguyen Hong An ;        18;     5.5

Test case 4 is OK
BUILD SUCCESSFUL (total time: 2 seconds)
```

```
run:
Enter TC (1-10): 5
OUTPUT:
Nothing to display(file is empty).

Expected:
Nothing to display(file is empty).

Test case 5 is OK
BUILD SUCCESSFUL (total time: 1 second)
```

```
run:
Enter TC (1-10): 5
OUTPUT:
He190011;       Nguyen Hong An ;        18;     5.5

Expected:
He190011;       Nguyen Hong An ;        18;     5.5

Test case 5 is OK
BUILD SUCCESSFUL (total time: 1 second)
```

```
Enter TC (1-10): 6
OUTPUT:
He190001;       Vu Van Huy ;    18;     9.5
He190006;       Bui Van Tan ;   18;     8.7
He190010;       Vu Minh Tuan ;  18;     8.0
He190005;       Pham Thanh Nam ;        22;     7.0
He190011;       Nguyen Hong An ;        18;     5.5

Expected:
He190001;       Vu Van Huy ;    18;     9.5
He190006;       Bui Van Tan ;   18;     8.7
He190010;       Vu Minh Tuan ;  18;     8.0
He190005;       Pham Thanh Nam ;        22;     7.0
He190011;       Nguyen Hong An ;        18;     5.5

Test case 6 is OK
BUILD SUCCESSFUL (total time: 2 seconds)
```

```
run:
Enter TC (1-10): 7
OUTPUT:
He190001;       Vu Van Huy ;    18;     9.5

Expected:
He190001;       Vu Van Huy ;    18;     9.5

Test case 7 is OK
BUILD SUCCESSFUL (total time: 1 second)
```

```
run:
Enter TC (1-10): 8
OUTPUT:
He190011;       Nguyen Hong An ;         18;      5.5

Expected:
He190011;       Nguyen Hong An ;         18;      5.5

Test case 8 is OK
BUILD SUCCESSFUL (total time: 1 second)
```

```
Enter TC (1-10): 9
OUTPUT:
He190001;       Vu Van Huy ;    18;     9.5;    passed
He190005;       Pham Thanh Nam ;        22;     7.0;    passed
He190006;       Bui Van Tan ;   18;     8.7;    passed
He190011;       Nguyen Hong An ;        18;     4.5;    Not passed
He190010;       Vu Minh Tuan ;  18;     8.0;    passed

Expected:
He190001;       Vu Van Huy ;    18;     9.5;    passed
He190005;       Pham Thanh Nam ;        22;     7.0;    passed
He190006;       Bui Van Tan ;   18;     8.7;    passed
He190011;       Nguyen Hong An ;        18;     4.5;    Not passed
He190010;       Vu Minh Tuan ;  18;     8.0;    passed

Test case 9 is OK
BUILD SUCCESSFUL (total time: 1 second)
```

```
Enter TC (1-10): 10
OUTPUT:
He190001;       Vu Van Huy ;    18;     9.5
He190006;       Bui Van Tan ;   18;     8.7
He190011;       Nguyen Hong An ;        18;     4.5
He190010;       Vu Minh Tuan ;  18;     8.0

Expected:
He190001;       Vu Van Huy ;    18;     9.5
He190006;       Bui Van Tan ;   18;     8.7
He190011;       Nguyen Hong An ;        18;     4.5
He190010;       Vu Minh Tuan ;  18;     8.0

Test case 10 is OK
BUILD SUCCESSFUL (total time: 2 seconds)
```