`Employee e1` does a set of projects. `Employee e2` also does all the projects did by `e1` except the first project, in place of which `e2` does another project. Write a program that defines two classes `Employee` and `Test`. Define copy constructor to create `e2` from `e1` in such a way that changing the values of instance variables of either `e2` or `e1` should not affect the other one. The code takes name of `e2` and new project done by `e2` as input. Complete the program as specified below.

- Class `Employee` that has the following members.
  - Private instance variables `String name` and `String[] projects` to store name and projects respectively
  - Define required constructor(s)
  - Accessor methods `getName( )` and `getProject( )` to get name of employee and project at specific index.
  - Mutator methods `setName( )` and `setProject( )` to set name of employee and project at specific index.
- Class `Test` that has the method `main` which does the following.
  - Two objects of `Employee e1` and `e2` are created. `e2` is created using `e1`
  - name of `e2` and second item bought by `c2` are updated by taking the input
  - name of `e1`, `e2` and first project done by `e1` and `e2` are printed

### Test Case 1

| Input | Expected Output | Actual Output |
| --- | --- | --- |
| Sneha PJ5 | Surya: PJ1<br>Sneha: PJ5 | |

### Test Case 2

| Input | Expected Output | Actual Output |
| --- | --- | --- |
| Rohan PJ7 | Surya: PJ1<br>Rohan: PJ7 | |

## Submission Results

### Test Case 1

| Input | Expected Output | Actual Output |
| --- | --- | --- |
| Megha PJ4 | Surya: PJ1<br>Megha: PJ4 | |

### Test Case 2

| Input | Expected Output | Actual Output |
| --- | --- | --- |
| Srija PJ0 | Surya: PJ1<br>Srija: PJ0 | |