|  |  |
| --- | --- |
| **Practicum Case** |  |
| COMP6122 | COMP6122001  Framework Layer Architecture |
| **Computer Science** | **O221-COMP6122-CT01-11** |
| ***Valid on*** *Odd Semester Year 2021/2022* | **Revision 00** |

**Learning Outcome**

* LO2 – apply design pattern in java
* LO3 – design object oriented in design pattern

**Topic**

* Session 11 – Behavioral Design Pattern III

## Sub Topics

* State
* Template Method

1. **State Pattern**

State is a behavioral design pattern that lets an object alter its behavior when its internal state changes. It appears as if the object changed its class. The State pattern suggests that you create new classes for all possible states of an object and extract all state-specific behaviors into these classes.

1. **Template Methods Pattern**

Template Method is a behavioral design pattern that defines the skeleton of an algorithm in the superclass but lets subclasses override specific steps of the algorithm without changing its structure. The Template Method pattern suggests that you break down an algorithm into a series of steps, turn these steps into methods, and put a series of calls to these methods inside a single template method. The steps may either be abstract or have some default implementation. To use the algorithm, the client is supposed to provide its own subclass, implement all abstract steps, and override some of the optional ones if needed (but not the template method itself).

## Soal

*Case*

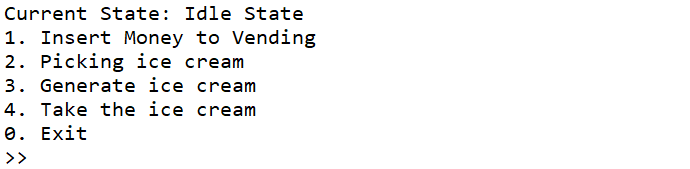
**Angelic Icy Cream**

AngelicIcy cream is one of the popular ice cream shops in Jakarta. At this moment, Angel as the manager of Angelic Icy cream has thought of an idea to improve the business. The idea is to create a vending machine that sells ice cream. You as the programmer of Angelic Icy cream, the manager asks you to create a prototype of a vending machine using Java. Below are the following details of the program:

There are 4 states on the vending machine, which are:

1. **Idle state**, which means the vending machine is idle.
2. **Picking ice cream state**, which means the vending machine is waiting for the user to choose what type of ice cream that he/she wants.
3. **Generate ice cream state**, which means the vending machine generating ice cream, and will ask what flavor (depends on the ice type of ice cream that he/she choose) that he/she wants.
4. **Ice cream is ready to take state**, which means the vending machine is finished generating ice cream, and the ice cream is ready to take.

* **Main Menu**
* **Show** the **menu** and **current state** of the vending machine.
* Ask the user to input **between** **0 and 5 (inclusive)**.

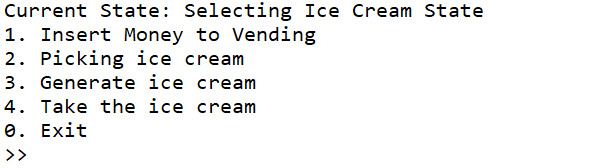


**Figure 1. Main Menu**

* If the user **choose 1** and the state is **idle** **state**, the state will become **picking/selecting ice cream state**.

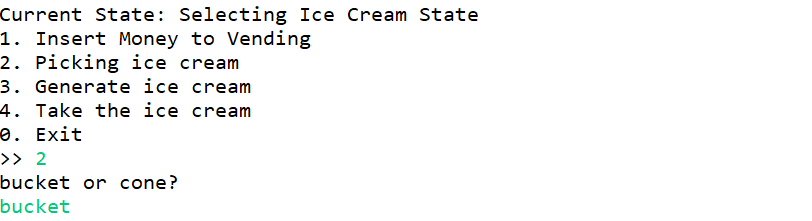


**Figure 2. User Choose 1 and State is Idle (Before)**

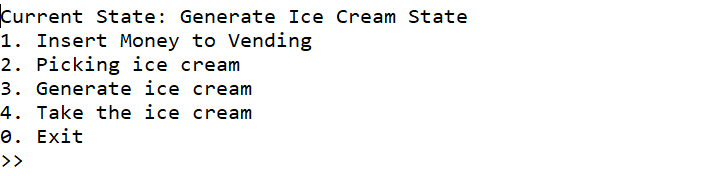


**Figure 3. User Choose 1 and State is Idle (After)**

* If the user **choose 2** and the state is **picking/selecting ice cream state.** Then, the vending machine will:
  + Ask the user to choose **type of ice cream**, is either ‘**bucket**’or ‘**cone**’.
  + After choosing the type of ice cream, the state will be turn into **generate ice cream state**.

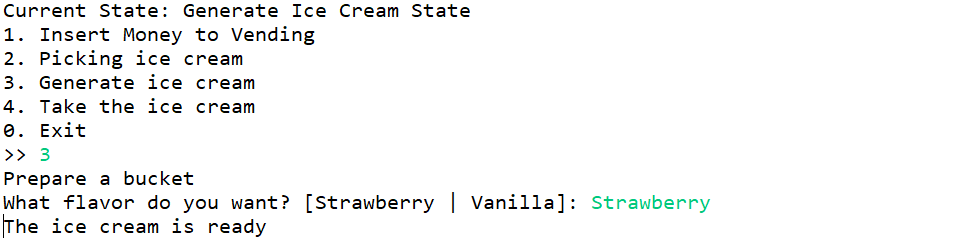


**Figure 4. User Choose 2 and State is Picking Ice Cream State (Before)**



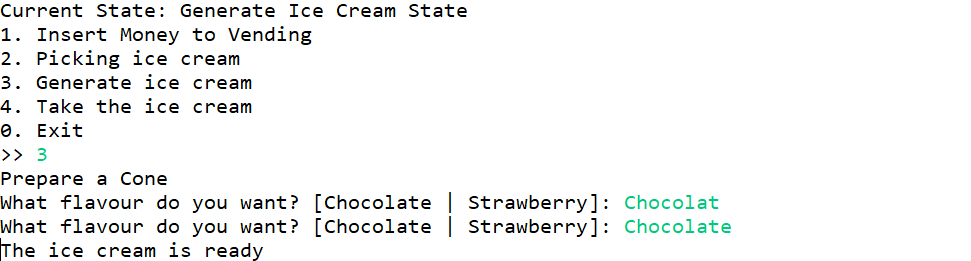
**Figure 5. User Choose 2 and State is Picking Ice Cream State (After)**

* If the user **choose 3** and the state is **generate ice cream state**. Then, the vending machine will:
  + Preparing the ice creamand ask the user to choose what **flavor** that the user wants. To prepare the ice cream and choosing the flavor of the ice cream, it all depends on the type of ice cream.
    - If the ice cream type is **bucket**, then the program will:
  + Prepare a **bucket**.
  + Ask the user to choose the flavor, is either “**Strawberry**” or “**Vanilla**”.



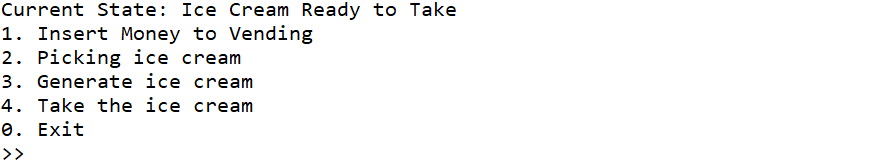
**Figure 6. Preparing and Choose Flavor (Bucket & Before)**

* + - If the ice cream type is **cone**, then the program will:
  + Prepare a **cone**.
  + Ask the user to choose the flavor, is either “**Strawberry**” or “**Chocolate**”.



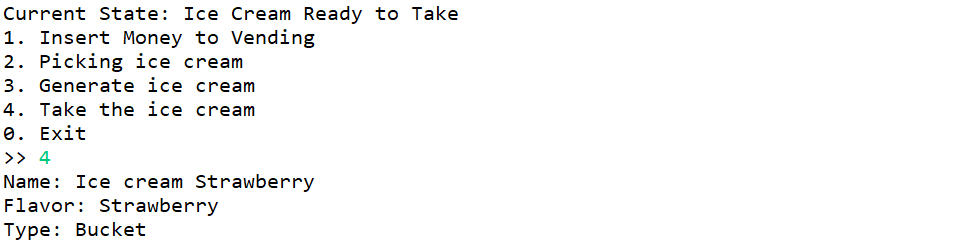
**Figure 7. Choose Flavor (Cone & Before)**

* + After that, the vending machine will **generate** **an ice cream** based on the type and the flavor of ice cream.
  + Lastly, the state will become **ice cream is ready to take state.**



**Figure 8 Preparing and Choose Flavor (after)**

* If the user **choose 4** and the state is **ice cream is ready to take state**.Then, the vending machine will:
  + Show the **name**, **flavor**,and **type** of the generated ice cream.
  + And thestatewill return to **idle state**.

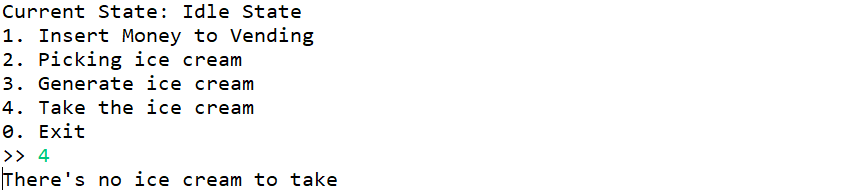


**Figure 9. Take Ice Cream (Before)**

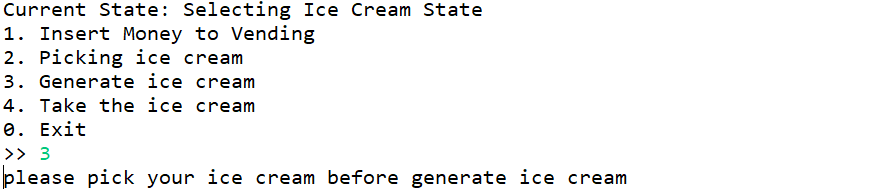


**Figure 10. Take Ice Cream (After)**

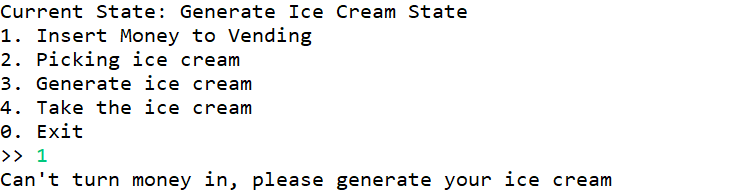
* + If the user **choose 0,** then the program will be **terminated**.
  + **Otherwise**, the vending machine will **print an error message.**



**Figure 11. Error Message (1)**



**Figure 12. Error Message (2)**



**Figure 13. Error Message (3)**