PHASE 4

SMART WATER MANAGEMENT

OBJECTIVE:

Our objective of this Development Part II is to provide a comprehensive and user friendly platform for real-time Smart Water Management aimed at enhancing the wastage of water .Through our advanced system, we aim to minimize the wastage of water ,ulmimately saving water.

PROGRAM:

Public class SmartWaterManagement

{

```
   Private boolean isOn;
   Private int waterLevel;
 Public SmartWaterManagement()
{
  isOn = false;
      waterLevel = 0;
 }
Public void turnOn()
{
       isOn = true;
       System.out.println("Water is now on.");
   }
 Public void turnOff()
{
   isOn = false;
   System.out.println("Water is now off.");
   }
 Public void fillWater(int amount)
{
 If (isOn)
{
```

```
    waterLevel += amount;
     System.out.println("Water level increased by " + amount+
liters.");
        }
Else
{
   System.out.println("Cannot fill water when the water is off.");
        }
    }
 Public void dispenseWater(int amount)
{
        If (isOn)
 {
      If (waterLevel >= amount)
 {
     waterLevel -= amount;
      System.out.println("Dispensing " + amount + " liters of
water.");
           }
 Else
 {
```

```
   System.out.println("Insufficient water for dispensing.");
            }
        }
Else
 {
  System.out.println("Cannot dispense water when the
managing water is off.");
        }
 Else
  Public int getWaterLevel()
{
    Return waterLevel;
   }
  Public boolean isOn()
 {
      Return isOn;
   }
 Public static void main(String[] args)
 {
   SmartWater Management = new SmartWater Management ();
management.turnOn();
```

```
   Management.fillWater(20);
    Management.dispenseWater(10);
   System.out.println("Current water level: " +
management.getWaterLevel());
   Management.turnOff();
    }
}
```

OUTPUT:

Waterlevel is now on.

Water level increased by 20 liters

Dispensing 10 liters of water

Current water level: 10

Water is now off.

CONCLUSION:

1. *You can turn it on or off using the turnOn and `turnoff' methods.*
2. *The `fillWater method allows adding Water if it's on.*
3. *The dispenseWater ` method Dispenses water if there's enough.*
4. *It provides access to the current water Level and on/off status.*
5. *The constructor initializes the water As off with no water.*
6. *It includes error messages for improper Actions.*
7. *Demonstrated in the 'main' method With on/off, filling, and dispensing.*

8. *Demonstrated in the 'main method With on/off, filling, and dispensing*
9. *A simple, encapsulated representation Of a smart water managing.*
10. *Offers essential functionality for Managing water levels.*