

```
1  using CustomProject.GameObjects;
2  using SplashKitSDK;
3
4  namespace CustomProject.StrategyDesign
5  {
6      public class AdvancedCollect : ICollectStrategy
7      {
8          public void Collect(Player player)
9          {
10             if (!player.IsDead)
11             {
12                 List<CollectibleGold> collectedGolds = new
13                     List<CollectibleGold>();
14                 List<Sword> collectedSwords = new List<Sword>();
15                 List<SpeedPotion> collectedSpeedPotion = new
16                     List<SpeedPotion>();
17                 List<JumpPotion> collectedJumpPotion = new List<JumpPotion>();
18
19                 foreach (CollectibleGold gold in player.GameMap.Golds)
20                 {
21                     if (SplashKit.BitmapCollision(player.Image, player.X,
22                         player.Y, gold.Image, gold.X, gold.Y))
23                     {
24                         collectedGolds.Add(gold);
25                     }
26                 }
27
28                 foreach (Sword sword in player.GameMap.Swords)
29                 {
30                     if (SplashKit.BitmapCollision(player.Image, player.X,
31                         player.Y, sword.Image, sword.X, sword.Y))
32                     {
33                         collectedSwords.Add(sword);
34                     }
35                 }
36
37                 foreach (SpeedPotion potion in player.GameMap.SpeedPotion)
38                 {
39                     if (SplashKit.BitmapCollision(player.Image, player.X,
40                         player.Y, potion.Image, potion.X, potion.Y))
41                     {
42                         collectedSpeedPotion.Add(potion);
43                     }
44                 }
45
46                 foreach (JumpPotion potion in player.GameMap.JumpPotion)
47                 {
48                     if (SplashKit.BitmapCollision(player.Image, player.X,
49                         player.Y, potion.Image, potion.X, potion.Y))
50                     {
51                         collectedJumpPotion.Add(potion);
52                     }
53                 }
54             }
55         }
56     }
```

```
        player.Y, potion.Image, potion.X, potion.Y))
44    {
45        collectedJumpPotion.Add(potion);
46    }
47}
48
49    foreach (SpeedPotion potion in collectedSpeedPotion)
50    {
51        player.GameMap.SpeedPotion.Remove(potion);
52        player.IncreaseSpeed = true;
53    }
54
55    foreach (JumpPotion potion in collectedJumpPotion)
56    {
57        player.GameMap.JumpPotion.Remove(potion);
58        player.IncreaseJump = true;
59    }
60
61    foreach (Sword sword in collectedSwords)
62    {
63        player.GameMap.Swords.Remove(sword);
64        player.CollectedSwords += 1;
65        player.Notify("SwordCollected");
66    }
67
68    foreach (CollectibleGold gold in collectedGolds)
69    {
70        player.GameMap.Golds.Remove(gold);
71        player.CollectedGolds += 1;
72        player.Notify("GoldCollected");
73    }
74}
75}
76}
77}
78}
79}
```