Normal çözüm: Soyut sınıf kullanarak. UML şemalarının çizimi öğrenciye bırakılmıştır.

```java
package ooc08.regular;
public abstract class Person {
   private String name;

   public Person(String name) {
      this.name = name;
   }
   public String getName() {
      return name;
   }
   public abstract void buyCandy();
   public abstract void buyCigarette();
}
```

```java
package ooc08.regular;
public class Kid extends Person {
   public Kid(String name) {
      super(name);
   }
   public void buyCigarette() {
      System.out.println("A kid cannot buy cigarettes");
      return;
   }
   public void buyCandy() {
      System.out.println("Wow. So many candies!");
   }
}
```

```java
package ooc08.regular;
public class YoungAdult extends Person {
   public YoungAdult(String name) {
      super(name);
   }
   public void buyCigarette() {
      System.out.println("Cigarettes are bought. You should consider quitting.");
   }
   public void buyCandy() {
      System.out.println("Candies are bought.");
   }

}
```

```java
package ooc08.regular;
public class Adult extends Person {
   public Adult(String name) {
      super(name);
   }
   public void buyCigarette() {
      System.out.println("Cigarettes are bought. Smoking Kills!");
   }
   public void buyCandy() {
      System.out.println("Candies are bought. Do not get fat.");
   }
}
```

```java
package ooc08.regular;
public class MainApp {
   public static void main( String[] args ) {
      Person[] people = new Person[3];
      people[0] = new Kid("Olcan Selçuk");
      people[1] = new YoungAdult("Fırat Ertemel");
      people[2] = new Adult("Yunus Emre Selçuk");
      for(Person person : people) {
         person.buyCandy();
         person.buyCigarette();
      }
   }
}
```

Aşağıdaki çözüm Strategy tasarım kalıbına göre arayüz kullanılarak yapılmıştır. UML şemalarının çizimi öğrenciye bırakılmıştır.

```java
package ooc08.strategy;

public interface IAge {
   public void buyCigarette();
   public void buyCandy();
}
```

```java
package ooc08.strategy;

public class Kid implements IAge {

   public void buyCigarette() {
      System.out.println("A kid cannot buy cigarettes");
      return;
   }

   public void buyCandy() {
      System.out.println("Wow. So many candies!");
   }

}
```

```java
package ooc08.strategy;

public class YoungAdult implements IAge {

   @Override
   public void buyCigarette() {
      System.out.println("Cigarettes are bought. You should consider quitting.");
   }
   @Override
   public void buyCandy() {
      System.out.println("Candies are bought.");
   }
}
```

```java
package ooc08.strategy;

public class Adult implements IAge {

   @Override
   public void buyCigarette() {
      System.out.println("Cigarettes are bought. Smoking Kills!");
   }

   @Override
   public void buyCandy() {
      System.out.println("Candies are bought. Do not get fat.");
   }

}
```

```java
package ooc08.strategy;

public class Person {
   private String name;
   private IAge age;

   public Person(String name, IAge age) {
      this.name = name;
      this.age = age;
   }
   public String getName() {
      return name;
   }
   public void buyCandies() {
      age.buyCandy();
   }
   public void buyCigars() {
      age.buyCigarette();
   }
   public static void main( String[] args ) {
      Person[] people = new Person[3];
      people[0] = new Person("Olcan Selçuk", new Kid());
      people[1] = new Person("Fırat Ertemel", new YoungAdult());
      people[2] = new Person("Yunus Emre Selçuk", new Adult());
      for(Person person : people) {
         person.buyCandies();
         person.buyCigars();
      }

   }


}
```