|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| old  client |  |  |  | target interface |
| **EnemyTank**  *(implements EnemyAtacker)* |  |  |  | EnemyAttacker  <interface> |
| fireWeapon(): void;  driveForward(): void;  assignDriver(String): void |  |  |  | fireWeapon(): void;  driveForward(): void;  assignDriver(String): void |
|  |  |  |  |  |
| new  client  Adaptee |  | Adapter |  | target interface |
|  |  |  |  |  |
|  |  | **EnemyRobotAdapter**  *(implements EnemyAtttacker)* |  |  |
| ***EnemyRobot*** |  | **EnemyRobot**  *(object inside + adapter construktor)* |  | **EnemyAttacker**  <interface> |
| smashWithHands(): void;  walkForward(): void;  reactToHuman(String): void; |  | fireWeapon(): void;  driveForward(): void;  assignDriver(String): void |  | fireWeapon(): void;  driveForward(): void;  assignDriver(String): void |
|  |  |  |  |  |
|  | public class EnemyRobotAdapter implements EnemyAttacker {  EnemyRobot **theRobot**;  public EnemyRobotAdapter(EnemyRobot newRobot) {  theRobot = newRobot;  }  public void fireWeapon() {  **theRobot**.**smashWithHands**();  }  public void driveForward() {  **theRobot**.**walkForward**();  }  public void assignDriver(String driverName) {  **theRobot**.**reactToHuman**(driverName);  }  } | | public class TestEnemyAttackers{  public static void main(String[] args){  EnemyTank rx7Tank = new EnemyTank();  EnemyRobot **fredTheRobot** = new EnemyRobot();  EnemyAttacker **robotAdapter** = new EnemyRobotAdapter(**fredTheRobot**);  System.out.println("The Robot with Adapter: ");  **robotAdapter**.assignDriver("Mark");  **robotAdapter**.driveForward();  **robotAdapter**.fireWeapon();  }  } | |