**package** cn.strategy;

/\*\*

\* 抽象策略角色

\* **@author** Administrator

\*

\*/

**public** **interface** Strategy {

**int** calculate(**int** a,**int** b);

}

**package** cn.strategy;

/\*\*

\* 具体策略模式

\* **@author** Administrator

\*

\*/

**public** **class** AddStrategy **implements** Strategy{

@Override

**public** **int** calculate(**int** a, **int** b) {

**return** a+b;

// **TODO** Auto-generated method stub

}

}

**package** cn.strategy;

**public** **class** SubtractStrategy **implements** Strategy {

@Override

**public** **int** calculate(**int** a, **int** b) {

// **TODO** Auto-generated method stub

**return** a-b;

}

}

**package** cn.strategy;

**public** **class** MultiplyStrategy **implements** Strategy {

@Override

**public** **int** calculate(**int** a, **int** b) {

// **TODO** Auto-generated method stub

**return** a\*b;

}

}

**package** cn.strategy;

**public** **class** DivideStrategy **implements** Strategy{

@Override

**public** **int** calculate(**int** a, **int** b) {

// **TODO** Auto-generated method stub

**return** a/b;

}

}

**package** cn.strategy;

/\*\*

\* 环境角色

\* **@author** Administrator

\*

\*/

**public** **class** Environment {

**public** Strategy getStrategy() {

**return** strategy;

}

**public** **void** setStrategy(Strategy strategy) {

**this**.strategy = strategy;

}

**private** Strategy strategy;

**public** Environment(Strategy strategy){

**this**.strategy = strategy;

}

**public** **int** calculate(**int** a,**int** b){

**return** strategy.calculate(a, b);

}

}

**package** cn.strategy;

**public** **class** Client {

**public** **static** **void** main(String[] args) {

Strategy addStrategy = **new** AddStrategy();

Environment environment = **new** Environment(addStrategy);

System.*out*.println(environment.calculate(3, 5));

Strategy subStrategy = **new** SubtractStrategy();

environment.setStrategy(subStrategy);

System.*out*.println(environment.calculate(3, 5));

Strategy multiplyStrategy = **new** MultiplyStrategy();

environment.setStrategy(multiplyStrategy);

System.*out*.println(environment.calculate(3, 5));

Strategy divideStrategy = **new** DivideStrategy();

environment.setStrategy(divideStrategy);

System.*out*.println(environment.calculate(3,5));

}

}