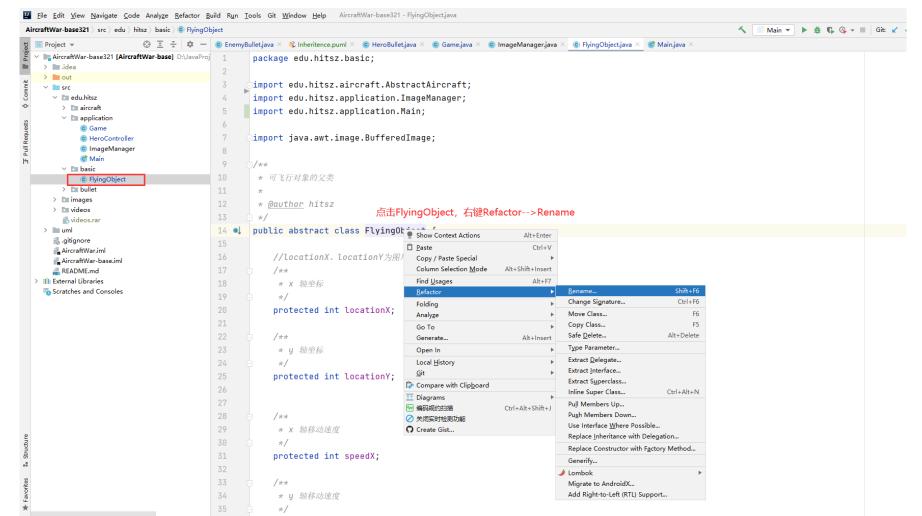


大家在开始实验二前先完成以下修改:

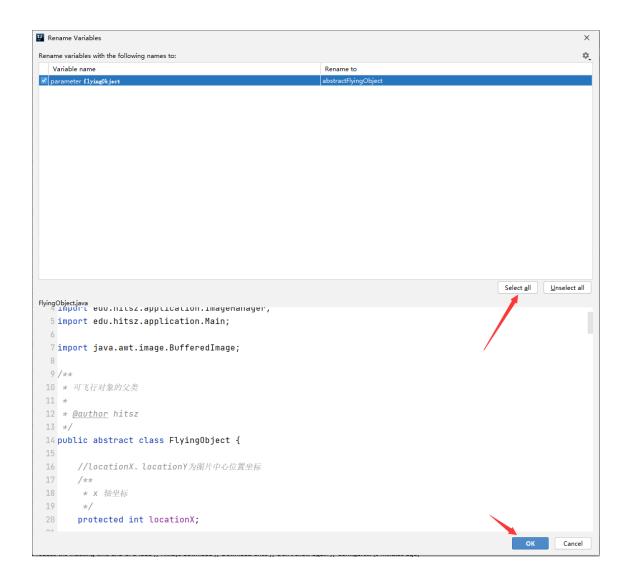
1、把basic下的FlyingObject改成AbstractFlyingObject;





```
可飞行对象的父类
  @author hitsz
 */
                                                 ■ Rename
                                                                                           ×
public abstract class FlyingObject " ⁴ {
                                                  Rename class 'edu.hitsz.basic.FlyingObject' and its usages to:
                                                   _AbstractFlyingObject
                                                                                           ₩
    //locationX、locationY为图片中心位置坐标
    /**
                                                    * X 轴坐标
                                                                          Rename inheritors
                                                   Rename tests
                                                  Rename variables
    protected int locationX;
                                                  Scope
                                                   Project Files
     * y 轴坐标
                                                   ?
                                                                    Refactor
                                                                             Preview
                                                                                       Cancel
```







可以看到类名、构造函数、用到这个类的地方都改过来了。

```
public abstract class AbstractFlyingObject {
     //locationX、locationY为图片中心位置坐标
      * X 轴坐标
      public AbstractFlyingObject() {
* @param abstractFlyingObject 撞击对方
* @return true: 我方被击中; false 我方未被击中
*/
public boolean crash(AbstractFlyingObject abstractFlyingObject) {
   // 缩放因子,用于控制 y轴方向区域范围
   int factor = this instanceof AbstractAircraft ? 2 : 1;
```



大家在开始实验二前先完成以下修改:

- 1、把basic下的FlyingObject改成AbstractFlyingObject;
- 2、把bullet下的AbstractBullet 改名为 BaseBullet; (因为这个类不是抽象类,所以叫AbstractBullet不那么合适,严谨一点改成 BaseBullet。修改方法跟刚才一模一样,用refactor。)



可以看到类名、构造函数、用到这个类的地方都改过来了。

```
* 也可以考虑不同类型的子弹
 * @author hitsz
public class BaseBullet extends AbstractFlyingObject {
    private int power = 10;
    public BaseBullet(int locationX, int locationY, int speedX, int speedY, int power) {
        super(locationX, locationY, speedX, speedY);
        this.power = power;
@Override
 * 通过射击产生子弹
 * @return 射击出的子弹List
public List<BaseBullet> shoot() {
    List<BaseBullet> res = new LinkedList<>();
    int x = this.getLocationX();
    int y = this.getLocationY() + direction*2;
    int speedX = 0;
    int speedY = this.getSpeedY() + direction*5;
    BaseBullet baseBullet;
    for(int i=0; i<shootNum; i++){</pre>
        // 子弹发射位置相对飞机位置向前偏移
        // 多个子弹横向分散
        baseBullet = new HeroBullet( locationX: x + (\underline{i}*2 - shootNum + 1)*10, y, speedX, speedY, power);
```



大家在开始实验二前先完成以下修改:

- 1、把basic下的FlyingObject改成AbstractFlyingObject;
- 2、把bullet下的AbstractBullet 改名为 BaseBullet;
- 3、按照刚才的命名修改UML类图,并把类图中原来构造函数的返回值都去掉。



AbstractFlyingObject 4 UML类图修改后 locationX:int locationY:int speedX:int speedY:int image:BufferedImage width:int height:int isValid:boolean AbstractFlyingObject(int locationX, int locationY, int speedX, int speedY) forward():void crash(AbstractFlyingObject flyingObject):boolean setLocation(double locationX, double locationY):void getLocationX():int getLocationY():int getSpeedY():int getImage():BufferedImage getWidth():int getHeight():int notValid():boolean vanish():void AbstractAircraft A BaseBullet axHp:int hp:int power:int AbstractAircraft(int locationX, int locationY, int speedX, int speedY, int hp) BaseBullet(int locationX, int locationY, int speedX, int speedY, int power) decreaseHp(int decrease):void forward():void getHp():int getPower():int shoot():List<BaseBullet> HeroAircraft MobEnemy C HeroBullet C EnemyBullet shootNum:int power:int direction:int MobEnemy(int locationX, int locationY, int speedX, int speedY, int hp) HeroBullet(int locationX, int locationY, EnemyBullet(int locationX, int locationY, forward():void HeroAircraft(int locationX, int locationY, int speedX, int speedY, int hp) int speedX, int speedY, int power) int speedX, int speedY, int power) shoot():List<BaseBullet> forward():void shoot():List<BaseBullet>