

<pre>public class AngryBird { public int getSpeed() { return 10; } public String getSound() { return "Squak!!"; } public Color getColor() { return Color.BLACK; } }</pre>	<pre>public class RedBird extends AngryBird { public Color getColor() { return Color.RED; } } public class BigBrotherBird extends RedBird { public int getSpeed() { return 5; } }</pre>
<pre>public class YellowBird extends AngryBird { public int getSpeed() { return 20; } public Color getColor() { return Color.YELLOW; } public void speedAttack() { Syso("I have the need..."); Syso("The need for speed!"); } }</pre>	<pre>public class BombBird extends AngryBird { public String getSound() { return "KABOOM!"; } public void explode() { Syso(getSound()); } }</pre>

- 1) On the back, draw the inheritance diagram (boxes with arrows) for the classes above.
- 2) For each class indicate the total number of methods the class has, as well as the number of overridden methods.

	AngryBird	RedBird	BigBrotherBird	YellowBird	BombBird
Total Methods					
Overridden Methods					

- 3) Cross out all the lines of code that are ILLEGAL. Write the output next to working lines of code.

<pre>AngryBird bird = new AngryBird(); Syso(bird.getSound()); bird.speedAttack(); RedBird bird2 = new RedBird(); Syso(bird2.getSpeed()); bird2.explode(); YellowBird bird3 = new YellowBird(); bird3.speedAttack();</pre>	<pre>BigBrotherBird b4 = new BigBrotherBird(); Syso(b4.getSound()); BombBird b5 = new BombBird(); b5.explode(); BigBrotherBird b6 = new RedBird(); AngryBird b7 = new BombBird(); b7.explode(); Syso(b7.getSound());</pre>
---	---

- 4) Write your own type of AngryBird class on the back!