

```

/**
 * This is the base pig player. Note that it is abstract, and so can't be
 * instantiated. Other pig players (e.g. <code>HumanPigPlayer</code>) must
 * extend this class.
 * @author Adam Smith
 * @version 1.0
 */

abstract public class PigPlayer {
    private String name; // player's name

    /**
     * The main constructor for <code>PigPlayer</code>.
     * @param name The <code>PigPlayer</code>'s name
     */
    public PigPlayer(String name) {
        this.name = name;
    }

    /**
     * Accessor for the <code>PigPlayer</code>'s name.
     * @return the <code>PigPlayer</code>'s name
     */
    public String getName() {
        return name;
    }

    /**
     * Alert the <code>PigPlayer</code> that its turn is beginning. This
method
     * needs to be implemented in the subclass (even if it is an empty
     * function).
     * @param myScore the player's current score
     * @param opponentsScore the opponent's current score
     */
    abstract public void beginTurn(int myScore, int opponentsScore);

    /**
     * Should the <code>PigPlayer</code> roll again? This method needs to
be
     * implemented in the subclass, taking the exact same arguments (even
though
     * some of them may be unused).
     * @param turnNumber which turn the player is on (0-indexed)
     * @param rollNumber which roll the player is on (0-indexed)
     * @param poolSize the number of points currently in the pool
     * @param myScore the number of points the player has already won
     * @param opponentsScore the number of points the opponent has already
won
     * @return true to roll again, false to stop
     */
    abstract public boolean decideIfShouldRoll(int turnNumber, int
rollNumber, int poolSize, int myScore, int opponentsScore);
}

```