```java
/* ***********************************
 *    PlayerI.java
 *       Interface for Player intellegence.
 * ***********************************/
package player;

import edu.neu.ccs.demeterf.demfgen.lib.List;
import gen.*;

/** Player Interface to reveal the various Agent implementations */
public interface PlayerI{
    /** Responsible for Derivative Creation */
    interface CreateAgentI{
        /** Returns a newly created derivative of a different type than already existing derivatives */
        Derivative createDerivative(Player player, List<Type> existing);
    }
    /** Responsible for Buying/selection */
    interface BuyAgentI{
        /** Returns a list of Derivatives to be purchased */
        List<Derivative> buyDerivatives(List<Derivative> forSale, double account);
    }
    /** Responsible for RawMaterial Delivery */
    interface DeliverAgentI{
        /** Returns RawMaterials for the given Derivatives */
        Derivative deliverRawMaterial(Derivative needRM);
    }
    /** Responsible for RawMaterial Finishing */
    interface FinishAgentI{
        /** Returns FinishedProduct for the given Derivative */
        FinishedProduct finishDerivative(Derivative needFinish);
    }

    /** Returns the Player's Simple Name */
    public String getName();
    /** Return the CreateAgent responsible for Derivative Creation */
    public CreateAgentI getCreateAgent();
    /** Return the BuyAgent responsible for Buying/selection */
    public BuyAgentI getBuyAgent();
    /** Return the DeliverAgent responsible for RawMaterial Delivery */
    public DeliverAgentI getDeliverAgent();
    /** Return the FinishAgent responsible for RawMaterial Finishing */
    public FinishAgentI getFinishAgent();
}
```