

## Class Player

java.lang.Object  
Player

Direct Known Subclasses:

Bot

```
public class Player
extends java.lang.Object
```

Klasse dient als Modell für einen Spieler.

Author:  
Philipp Dobieszewski

### Constructor Summary

Constructors
Constructor and Description
<b>Player()</b> Konstruktor gibt dem Spieler eine Startguthaben.

### Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method and Description	
int	<b>bet</b> (int bet)	Setzt einen Einsatz.
java.lang.String	<b>checkCards</b> (List<Card> cCards)	Überprüft Karten auf Regeln.
void	<b>fold</b> ()	Setzt Spielstatus auf false.
int	<b>getBalance</b> ()	Gibt Spielerguthaben zurück.
Card[]	<b>getPocket</b> ()	Gibt Karten des Spielers zurück.
boolean	<b>isBBlind</b> ()	Gibt den Big-Blind-Status zurück.
boolean	<b>isInGame</b> ()	Gibt den Spielstatus zurück.

boolean	<b>isSBlind()</b> Gibt den Small-Blind-Status zurück.
<b>List&lt;Card&gt;</b>	<b>pocketToList()</b> Konvertiert <b>Card</b> -Array zu <b>List</b> .
void	<b>setBalance</b> (int balance) Setzt Spielerguthaben.
void	<b>setBBlind</b> (boolean bBlind) Setzt den Big-Blind-Status.
void	<b>setInGame</b> (boolean inGame) Setzt den Spielstatus.
void	<b>setPocket</b> ( <b>List&lt;Card&gt;</b> pocket) Setzt Spielerkarten.
void	<b>setSBlind</b> (boolean sBlind) Setzt den Small-Blind-Status.
<b>Methods inherited from class java.lang.Object</b>	
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait	

### Constructor Detail

#### Player

```
public Player()
```

Konstruktor gibt dem Spieler eine Startguthaben.

### Method Detail

#### bet

```
public int bet(int bet)
```

Setzt einen Einsatz.

##### Parameters:

bet - Einsatz

##### Returns:

Einsatz

#### getPocket

```
public Card[] getPocket()
```

Gibt Karten des Spielers zurück.

##### Returns:

Karten

### setPocket

```
public void setPocket(List<Card> pocket)
```

Setzt Spielerkarten.

#### Parameters:

pocket - Spielerkarten

### getBalance

```
public int getBalance()
```

Gibt Spielerguthaben zurück.

#### Returns:

Guthaben

### setBalance

```
public void setBalance(int balance)
```

Setzt Spielerguthaben.

#### Parameters:

balance - Guthaben

### isSBlind

```
public boolean isSBlind()
```

Gibt den Small-Blind-Status zurück.

#### Returns:

Status

### setSBlind

```
public void setSBlind(boolean sBlind)
```

Setzt den Small-Blind-Status.

#### Parameters:

sBlind - Small-Blind

### isBBlind

```
public boolean isBBlind()
```

Gibt den Big-Blind-Status zurück.

**Returns:**

Status

### setBBlind

```
public void setBBlind(boolean bBlind)
```

Setzt den Big-Blind-Status.

**Parameters:**

bBlind - Big-Blind

### isInGame

```
public boolean isInGame()
```

Gibt den Spielstatus zurück.

**Returns:**

Spielstatus

### setInGame

```
public void setInGame(boolean inGame)
```

Setzt den Spielstatus.

**Parameters:**

inGame - Spielstatus

### fold

```
public void fold()
```

Setzt Spielstatus auf false.

### pocketToList

```
public List<Card> pocketToList()
```

Konvertiert Card-Array zu List.

**Returns:**

List aus Cards

### checkCards

```
public java.lang.String checkCards(List<Card> cCards)
```

Überprüft Karten auf Regeln.

**Parameters:**

cCards - Karten

Returns:

Regel

[PACKAGE](#) **[CLASS](#)** [USE](#) [TREE](#) [DEPRECATED](#) [INDEX](#) [HELP](#)

**[PREV CLASS](#)** **[NEXT CLASS](#)** [FRAMES](#) [NO FRAMES](#) [ALL CLASSES](#)

[SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)