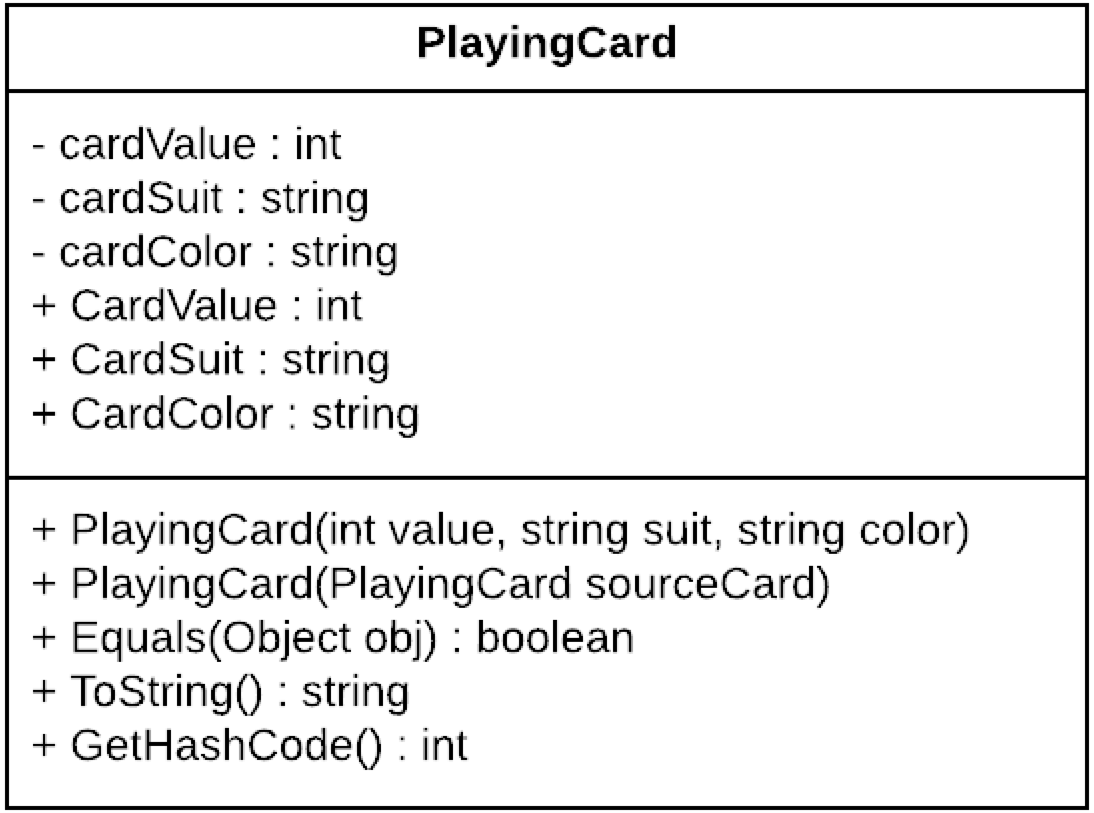
Inheritance Examples Using the PlayingCard Class

PlayingCard Class

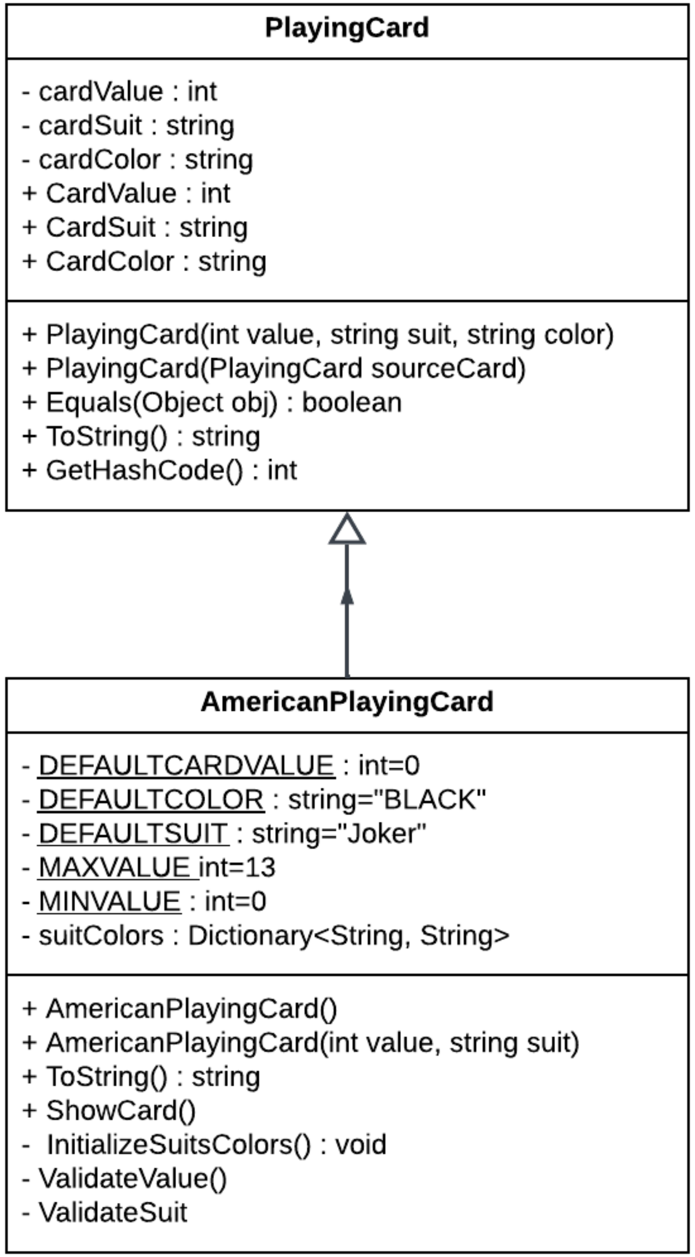


```
PlayingCard
- cardValue : int
- cardSuit : string
- cardColor : string
+ CardValue : int
+ CardSuit : string
+ CardColor : string
+ PlayingCard(int value, string suit, string color)
+ PlayingCard(PlayingCard sourceCard)
+ Equals(Object obj) : boolean
+ ToString() : string
+ GetHashCode() : int
```

A PlayingCard has:

- A value
- A suit
- A color
- A 3-arg constructor that takes a value, suit and color
- A Copy Constructor to make a “deep copy” of a PlayingCard from another PlayingCard
- Properties to allow access to the data in an object
- A **ToString()** method to convert the values to a String
- An **Equals()** method to determine if two PlayingCard objects have the same value, suit and color
- A **GetHashCode()** method to generate a HashCode

AmericanPlayingCard Class



PlayingCard

- cardValue : int
- cardSuit : string
- cardColor : string
+ CardValue : int
+ CardSuit : string
+ CardColor : string

+ PlayingCard(int value, string suit, string color)
+ PlayingCard(PlayingCard sourceCard)
+ Equals(Object obj) : boolean
+ ToString() : string
+ GetHashCode() : int

AmericanPlayingCard

- DEFAULTCARDVALUE : int=0
- DEFAULTCOLOR : string="BLACK"
- DEFAULTSUIT : string="Joker"
- MAXVALUE int=13
- MINVALUE : int=0
- suitColors : Dictionary<String, String>

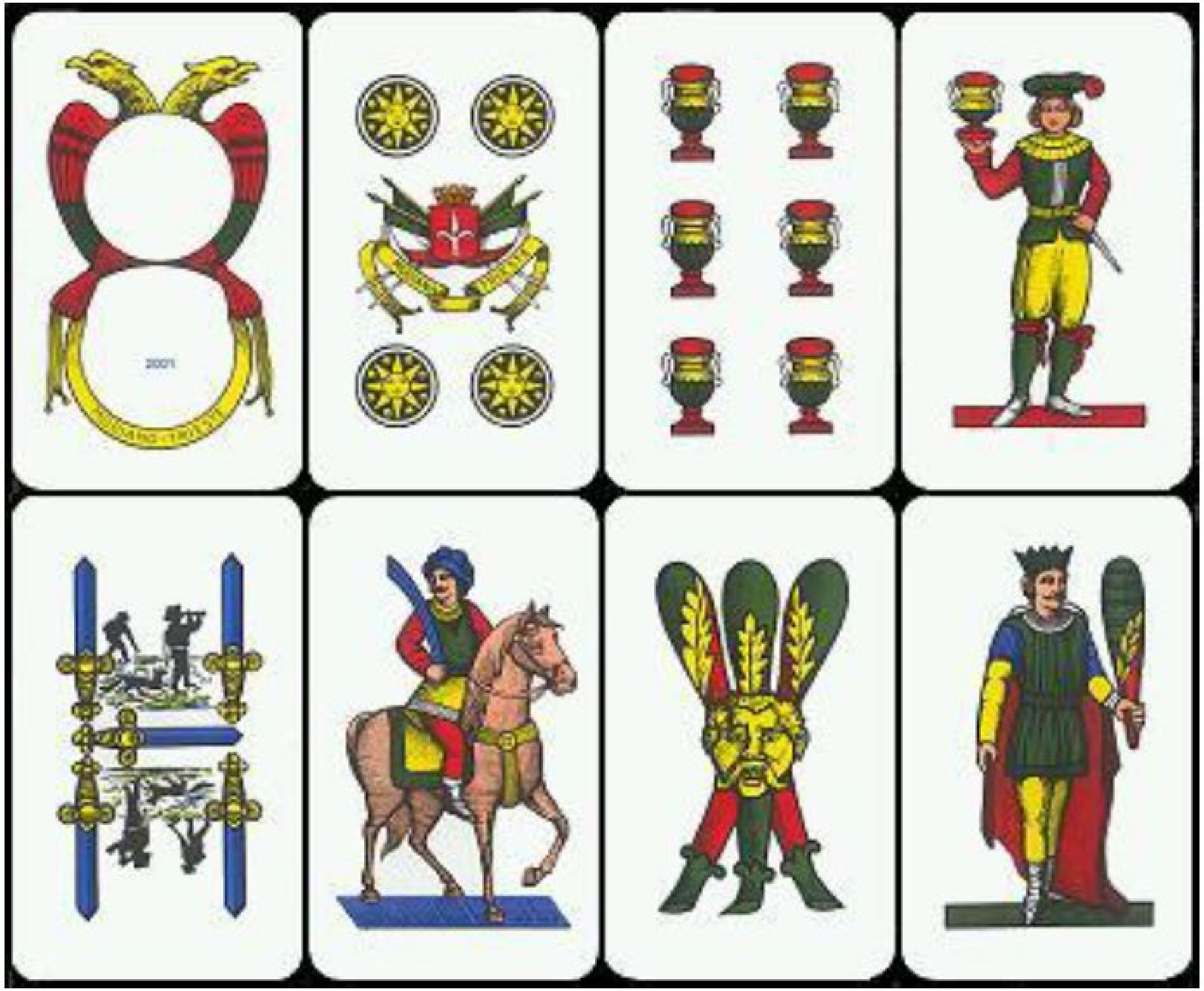
+ AmericanPlayingCard()
+ AmericanPlayingCard(int value, string suit)
+ ToString() : string
+ ShowCard()
- InitializeSuitsColors() : void
- ValidateValue()
- ValidateSuit

An AmericanPlayingCard is a PlayingCard with:

- A maximum value of 13 (King)
- A minimum value of 0 (Joker)
- Default value of 0
- Default suit of “Joker”
- Default color of “Black”
- A `suitColors` Dictionary() used to relate suits and colors

  Suits/Colors are: SPADES/BLACK, CLUBS/BLACK,
  HEARTS/RED, DIAMONDS/RED
- A 2-arg constructor that takes a value, suit
- A `ToString()` method to convert the values to a String
- An `InitializeSuitsColors()` method to initialize the `suitColors` Dictionary valid values
- `ValidateValue()` method to ensure a valid value is stored
- `ValidateSuit()` method to ensure a valid suit is stored

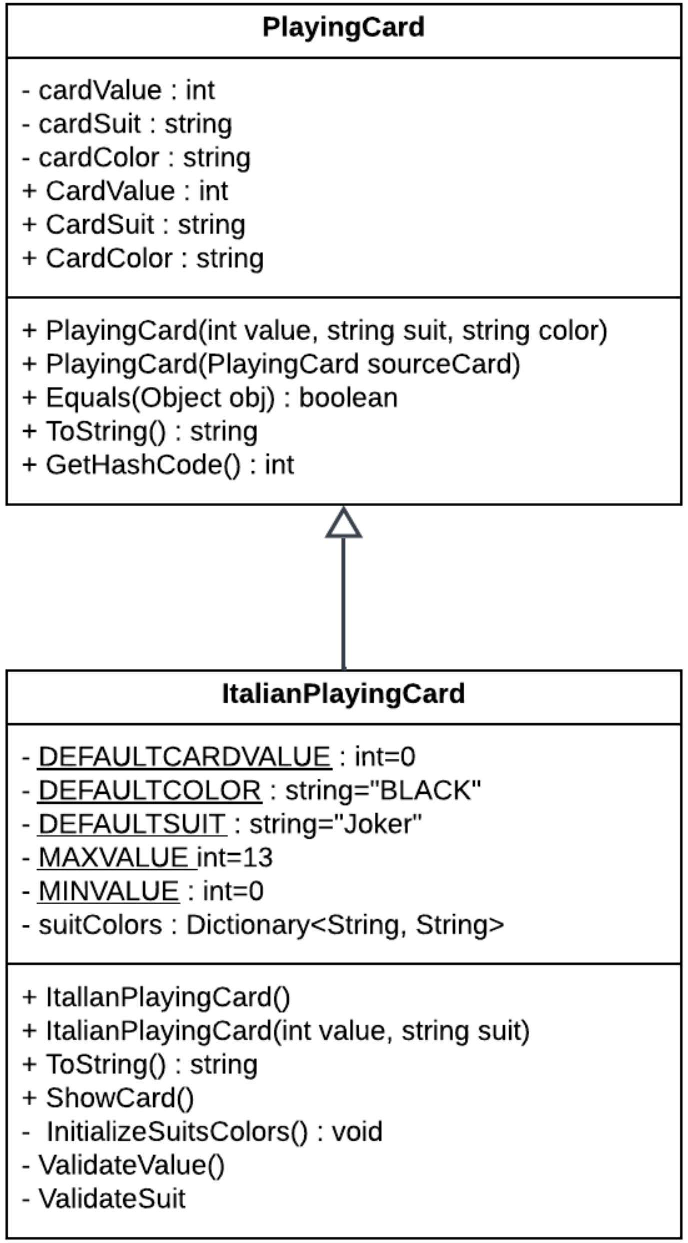
Italian *(Scopa)* Playing Cards



- Values:
  - 1 to 7, 10, ”Fante”, ”Cavallo”, ”Re”

- Suits/Colors are:
  - COINS - Yellow
  - CUPS - Blue
  - SWORDS - Red
  - BATONS - Black

ItalianPlayingCard Class



PlayingCard

- cardValue : int
- cardSuit : string
- cardColor : string
+ CardValue : int
+ CardSuit : string
+ CardColor : string

+ PlayingCard(int value, string suit, string color)
+ PlayingCard(PlayingCard sourceCard)
+ Equals(Object obj) : boolean
+ ToString() : string
+ GetHashCode() : int

ItalianPlayingCard

- DEFAULTCARDVALUE : int=0
- DEFAULTCOLOR : string="BLACK"
- DEFAULTSUIT : string="Joker"
- MAXVALUE int=13
- MINVALUE : int=0
- suitColors : Dictionary<String, String>

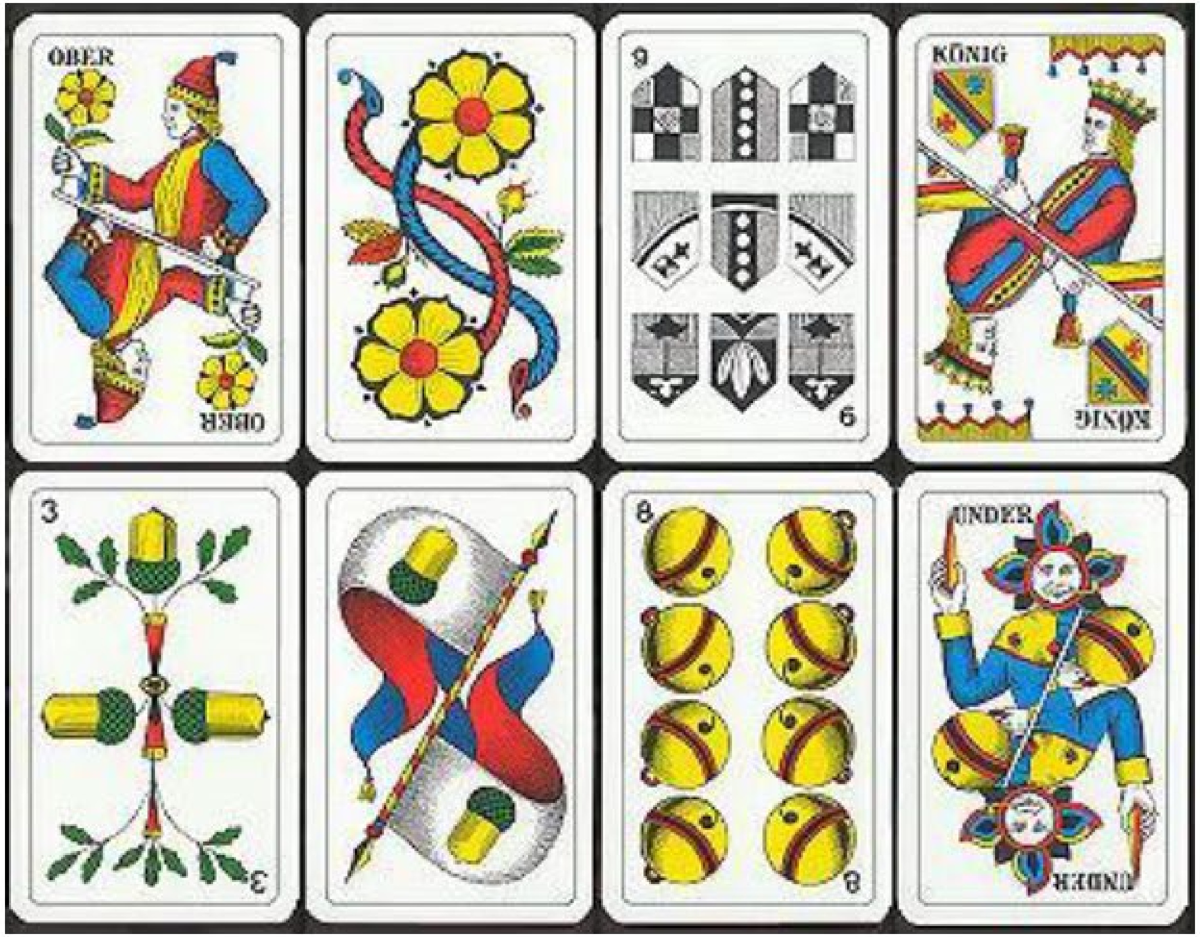
+ ItallanPlayingCard()
+ ItalianPlayingCard(int value, string suit)
+ ToString() : string
+ ShowCard()
- InitializeSuitsColors() : void
- ValidateValue()
- ValidateSuit

An ItalianPlayingCard is a PlayingCard with:

- A maximum value of 13 (Re) (There are no 8's or 9's)
- A minimum value of 0 (Joker)
- Default value of 0
- Default suit of "Joker"
- Default color of "Black"
- A `suitColors` Dictionary() used to relate suits and colors

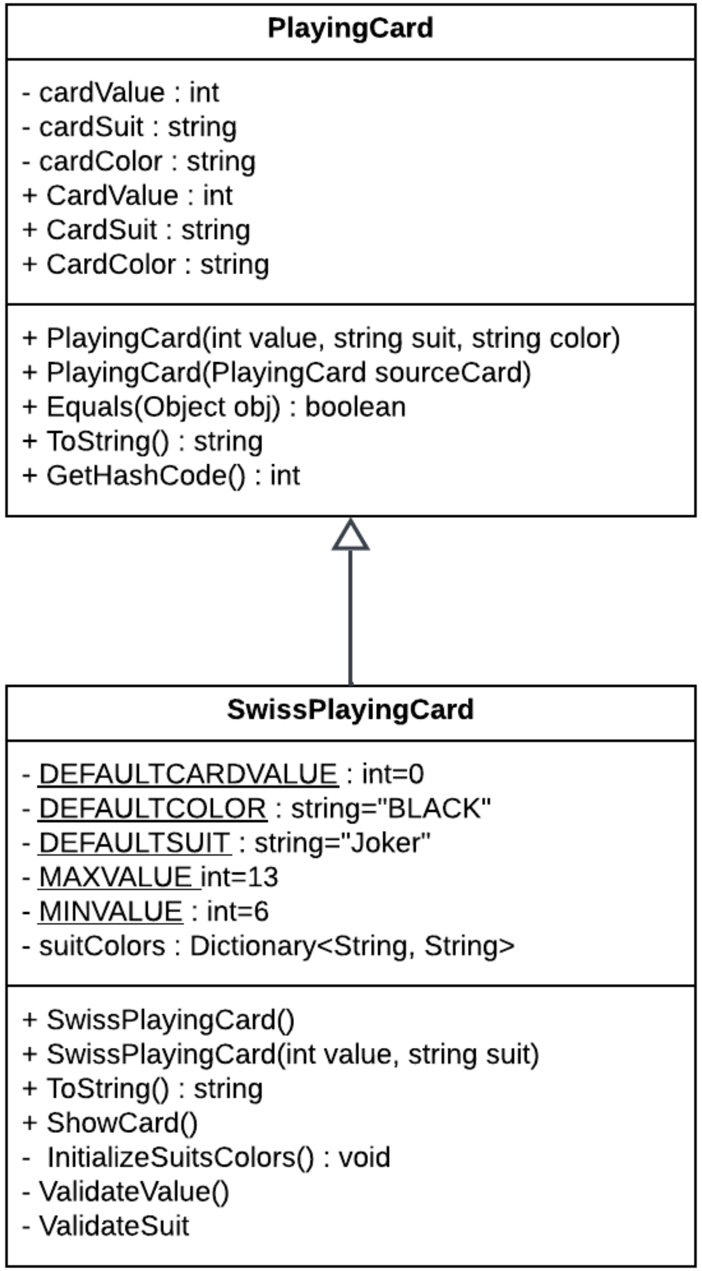
  Suits/Colors are: COINS/YELLOW, CUPS/BLUE,
  SWORDS/RED, BATONS/BLACK
- A 2-arg constructor that takes a value, suit
- A `ToString()` method to convert the values to a String
- An `InitializeSuitsColors()` method to initialize the `suitColors` Dictionary valid values
- `ValidateValue()` method to ensure a valid value is stored
- `ValidateSuit()` method to ensure a valid suit is stored

Swiss Playing Cards



- Values:
  - 6 to 9, “Banner”, ”Under”, ”Ober”, "König"; 1 - “As”
- Suits/Colors are:
  - BALLS - Yellow
  - ACORNS - Green
  - ROSES - Red
  - SHIELDS - Black

SwissPlayingCard Class



PlayingCard

- cardValue : int
- cardSuit : string
- cardColor : string
+ CardValue : int
+ CardSuit : string
+ CardColor : string

+ PlayingCard(int value, string suit, string color)
+ PlayingCard(PlayingCard sourceCard)
+ Equals(Object obj) : boolean
+ ToString() : string
+ GetHashCode() : int

SwissPlayingCard

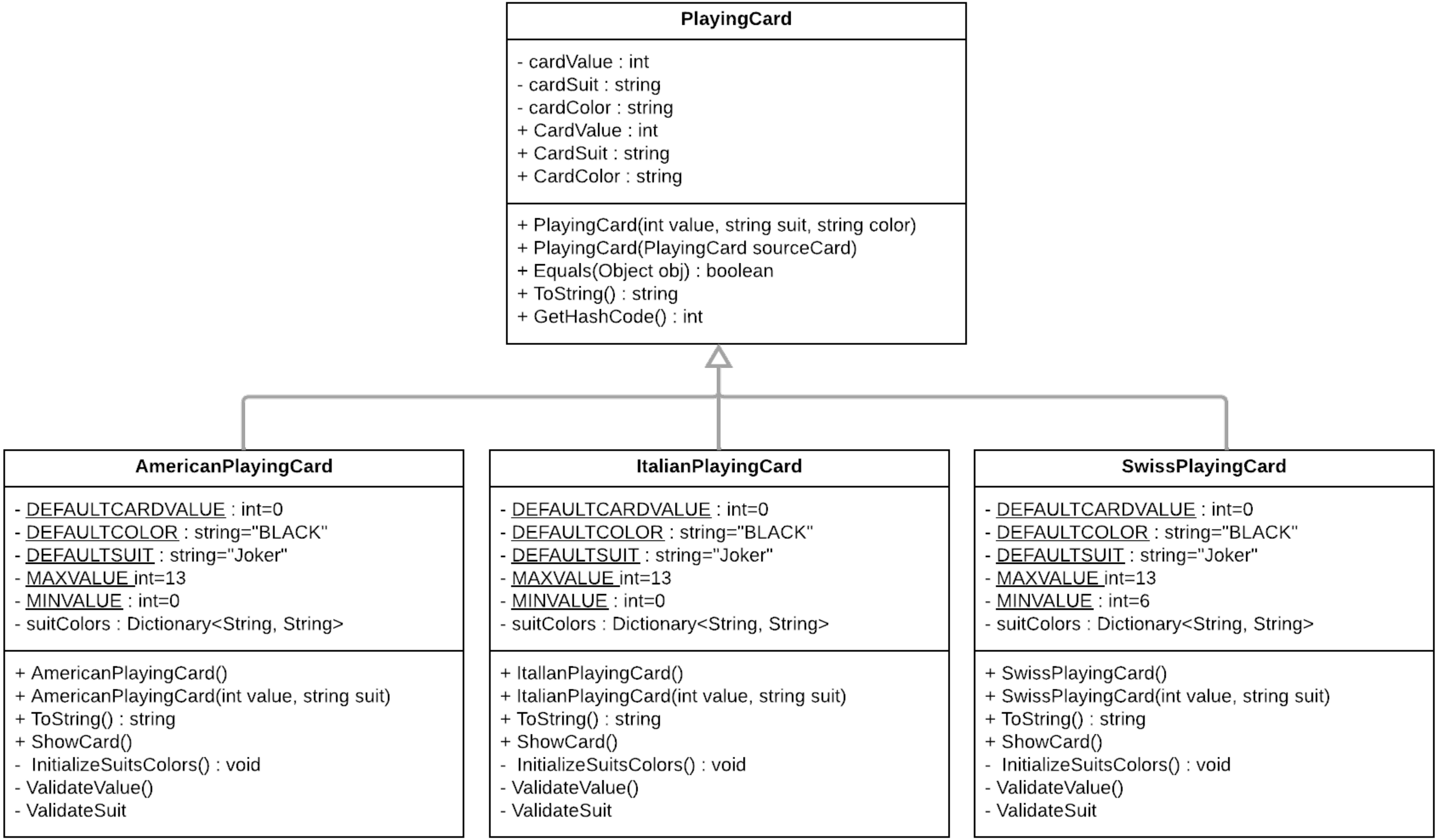
- DEFAULTCARDVALUE : int=0
- DEFAULTCOLOR : string="BLACK"
- DEFAULTSUIT : string="Joker"
- MAXVALUE int=13
- MINVALUE : int=6
- suitColors : Dictionary<String, String>

+ SwissPlayingCard()
+ SwissPlayingCard(int value, string suit)
+ ToString() : string
+ ShowCard()
- InitializeSuitsColors() : void
- ValidateValue()
- ValidateSuit

An SwissPlayingCard is a PlayingCard with:

- A maximum value of 13 (König) (There are no 2’s or 5’s)
- A minimum value of **6** (Joker)
- Default value of 0
- Default suit of “Joker”
- Default color of “Black”
- A `suitColors` Dictionary() used to relate suits and colors

  Suits/Colors are: BALLS/YELLOW, ACORNS/GREEN,
  ROSES/RED, SHIELDS/BLACK
- A 2-arg constructor that takes a value, suit
- A `ToString()` method to convert the values to a String
- An `InitializeSuitsColors()` method to initialize the `suitColors` Dictionary valid values
- `ValidateValue()` method to ensure a valid value is stored
- `ValidateSuit()` method to ensure a valid suit is stored



PlayingCard

- cardValue : int
- cardSuit : string
- cardColor : string
+ CardValue : int
+ CardSuit : string
+ CardColor : string

+ PlayingCard(int value, string suit, string color)
+ PlayingCard(PlayingCard sourceCard)
+ Equals(Object obj) : boolean
+ ToString() : string
+ GetHashCode() : int

AmericanPlayingCard

- DEFAULTCARDVALUE : int=0
- DEFAULTCOLOR : string="BLACK"
- DEFAULTSUIT : string="Joker"
- MAXVALUE int=13
- MINVALUE : int=0
- suitColors : Dictionary<String, String>

+ AmericanPlayingCard()
+ AmericanPlayingCard(int value, string suit)
+ ToString() : string
+ ShowCard()
- InitializeSuitsColors() : void
- ValidateValue()
- ValidateSuit

ItalianPlayingCard

- DEFAULTCARDVALUE : int=0
- DEFAULTCOLOR : string="BLACK"
- DEFAULTSUIT : string="Joker"
- MAXVALUE int=13
- MINVALUE : int=0
- suitColors : Dictionary<String, String>

+ ItallanPlayingCard()
+ ItalianPlayingCard(int value, string suit)
+ ToString() : string
+ ShowCard()
- InitializeSuitsColors() : void
- ValidateValue()
- ValidateSuit

SwissPlayingCard

- DEFAULTCARDVALUE : int=0
- DEFAULTCOLOR : string="BLACK"
- DEFAULTSUIT : string="Joker"
- MAXVALUE int=13
- MINVALUE : int=6
- suitColors : Dictionary<String, String>

+ SwissPlayingCard()
+ SwissPlayingCard(int value, string suit)
+ ToString() : string
+ ShowCard()
- InitializeSuitsColors() : void
- ValidateValue()
- ValidateSuit