**Code interview question:**

1. Work with the language you are most familiar with, but preferably the language which is the focus of the hiring.
2. The code you produce should be as close to “production quality” as possible, time permitting. This means you should document all assumptions, perform all tests, linting, etc. Tests should be extensible.
3. Code should include unit tests with > 90% test coverage using either unittest or pytest.
4. Code should be linted and formatted (pylint and black are recommended).
5. Code should be documented according to Google's Python style guide (https://google.github.io/styleguide/pyguide.html).
6. Code should demonstrate your ability in these areas:

* importing libraries
* using OO design

1. Defer to a “standard” deck of cards.
2. Post the completed product to a public *Github* and send the link to recruiter.

**Create a card game which supports all of the operations below.**

1. Shuffle cards in the deck: randomly mix the cards in the card deck, and return a whole deck of cards with a mixed order.
2. Get a card from the top of the deck: get one card from the top of the card deck, return a card, and if there is no card left in the deck, return error or exception.
3. Sort cards: Sort cards in ascending order by suit and rank (ace high).

i.e. If the deck contains cards with the following order:

(Spades, 2), (Diamonds, 5), (Spades, King), (Hearts, 3), (Clubs, Ace)

Sort cards ([Spades, Diamonds, Hearts, Clubs]) will return the cards with following order:

(Spades, 2), (Spades, King), (Diamonds, 5), (Hearts, 3), (Clubs, Ace)

1. Determine winners: 2 players play the game. They will draw 3 cards by taking turns.

Whoever has the high score wins the game.

Suit point number calculation: Clubs = 4, Hearts = 3, Diamonds = 2, Spades = 1

The winning value is calculated by **suit point number \* number** in the card.