

# Gramy w Oczko, czyli Black Jack!

## Moduł 1



# Obowiązkowo



- ✓ **Wiem jak tworzyć klasy**
- ✓ **Potrafię łączyć obiekty ze sobą(kompozycja)**
- ✓ **Wiem czym jest self oraz `__init__`**
- ✓ **Potrafię tworzyć wyjątki oraz je obsługiwać**

# Opcjonalnie

- ✓ Możesz zerknąć na funkcję `product` z `itertools`
- ✓ Najpewniej można będzie użyć `valrus := operator`
- ✓ Rozumiem na czym polega enkapsulacja i zmienne prywatne
- ✓ Wiem czym różnią się metody statyczne od “zwykłych”





## Projekt 1

# Symulator gry w Black Jack'a!

# Opis zadania

Black Jack to gra w której gracz(Ty) mierzy się z krupierem. Co będzie Ci potrzebne?

Wyobraź sobie stos kart.. każda karta posiada swoją wartość od 2 do ASa (pamiętaj o K, J, Q!). Każda z kart posiada też jeden z czterech kolorów kier, karo, trefl i pik.

## **Gra się rozpoczyna..**

Krupier tasuje karty, a następnie wydaje dwie karty graczowi i dwie sobie. Gracz widzi swoje karty, ale nie wie jakie karty posiada krupier. Kolejną decyzją gracza jest czy bierze kolejną kartę od gracza czy pasuje.

## **Kiedy gra się kończy?**

Koniec nastąpi wtedy kiedy gracz spasuje lub wtedy gdy suma jego kart przekroczy 21.

## **Kto wygrywa?**

Krupier wygra wtedy gdy gracz spasuje, a liczba jego punktów będzie mniejsza niż 21 oraz większa od ilości punktów zdobytych przez gracza,



# Jak liczyć karty?

My tutaj doskonalimy się w Pythonie.. wiemy, że Black Jack jest grą odrobinę bardziej skomplikowaną niż to tutaj ujęliśmy, ale wydaje się nam, że taka ilość pracy będzie i tak wystarczająca.. Natomiast jeśli macie chęć później to rozwinąć to zapraszamy!

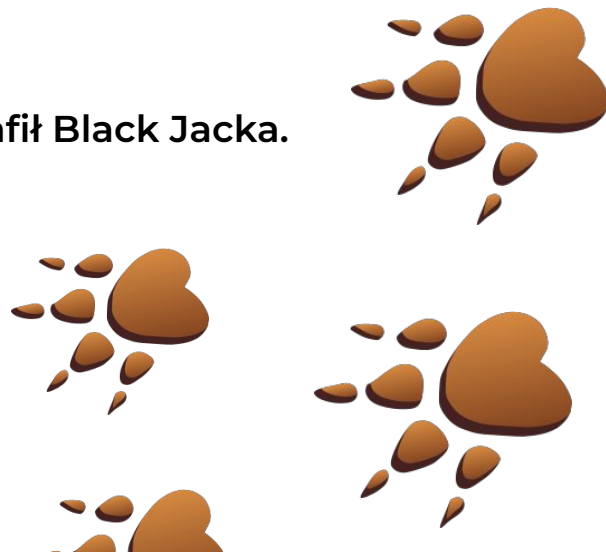
## Jak liczyć karty?

Każda karta ma swoją wartość zgodną z jego liczbą. Dwójka to dwa oczka, szóstka to 6. Karty, które są figurami (król, dama, walet) zawsze liczą się jako 10 oczek.

## A co z Asem?

Jeśli gracz w rozdaniu trafi 2 asy, to uznajemy, że posiada 21. Trafił Black Jacka. Jeśli gracz posiada asa i figurę to uznajemy, że posiada 21.

Jeśli gracz posiada na ręce trzy karty np.  
Króla, szóstkę i asa.. to asa liczymy jako 1 punkt.  
Czyli suma oczek wynosi 17 (10 + 6 + 1)



# Treść zadania

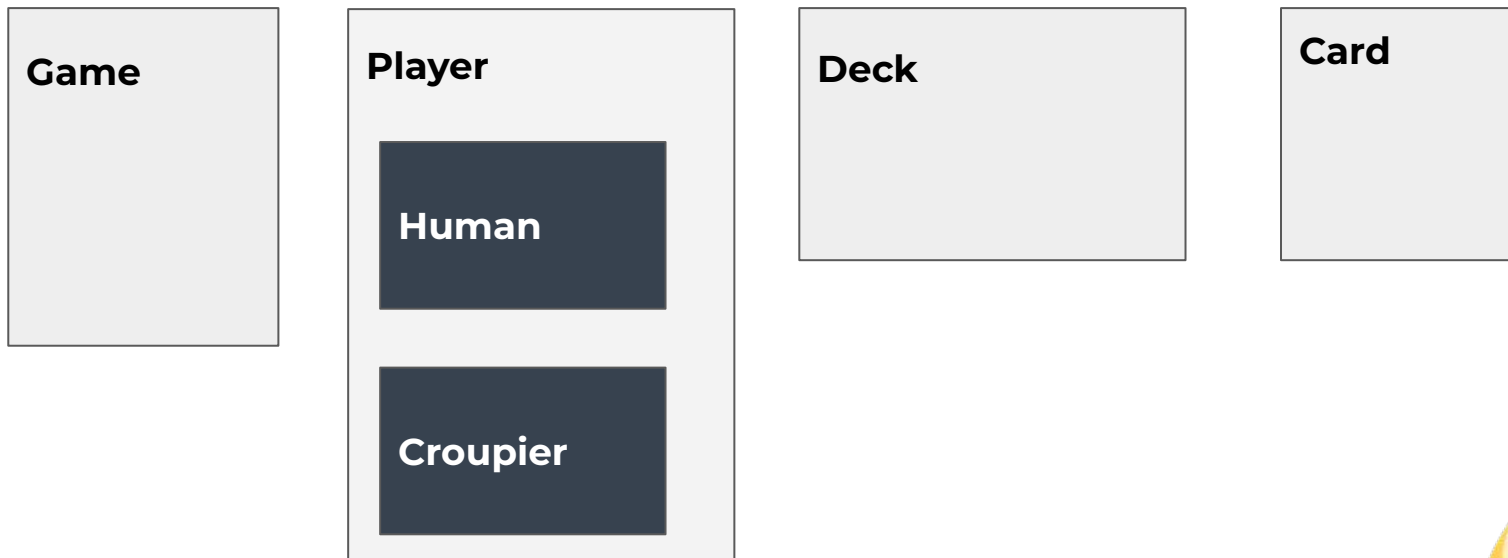
Dobra! Myślę, że możemy uznać, że zasady Black Jacka są dla Ciebie jasne. Tak naprawdę projekt ten polega na tym aby taką grę sobie teraz przygotować w pełni obiektowo.

Na początku połączenie tych wszystkich obiektów może stanowić dla Ciebie problem. Jest to w pełni zrozumiałe.. dlatego też przygotowaliśmy dla Ciebie małą podpowiedź.

**Bardzo zachęcamy do tego aby w tych projektach pisać testy.**



# Podpowiedź - podejście 1





# Podpowiedź - podejście 2

**Game**

**Player**

**Hand**

**Card**

**Deck**



# Podpowiedź - wyjątki

**GameOverException**

Gra się kończy z jakiegoś powodu..



**GameOverPlayerException**

Player popełnił błąd, wygrał krupier

**GameOverCroupierException**

Krupier popełnił błąd, wygrał gracz



**Jak dalej rozwinąć ten projekt?**

## Dalsze pomysły

- ✓ Dorób opcję liczenia kart, by gracz miał podpowiedzi i wiedział czy warto grać dalej czy spasować.
- ✓ Pomyśl o tym by zrobić interfejs graficzny dla BlackJacka np. używając tkintera.
- ✓ Możesz np. dorobić mechanizm obsługi żetonów i stawiania zakładów, a następnie sprawić by gra trwała aż nie ograsz krupiera :-)
- ✓ Blackjack ma kilka dodatkowych zasad np. dotyczących splitowania kart itp. Można zawsze coś z tego zaimplementować.