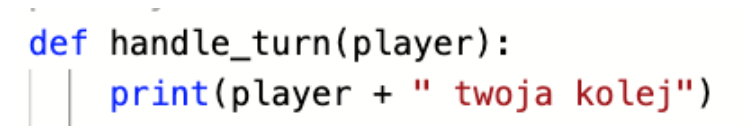
**Obsługa tury gracza.**

Jak dobrze wiesz, gra kółko i krzyżyk jest grą turową, w której gracze naprzemiennie uzupełniają pola na planszy. Musimy więc napisać skrypt, który pozwoli nam uzupełniać nasze pola odpowiednimi znakami.

Stworzymy więc funkcję handle\_turn(player). Musimy w niej wyświetlić informację, którego gracza dotyczy dana tura. Skorzystamy więc z funkcji print, która wyświetli “nazwę gracza” i odpowiedni tekst. Definiując więc działanie naszej funkcji wpisujemy print(player + " twoja kolej")



Zawrzemy w niej również zmienną position, która będzie przechowywać wartość wpisaną przez użytkownika. Skorzystamy więc z funkcji input. Zrób to w ten sposób:

image

W efekcie otrzymamy możliwość umieszczenia swojego symbolu poprzez wykorzystanie funkcji input, a nasz system przed każdą taką operacją wyświetli tekst “Wybierz pozycję od 1 do 9:”.

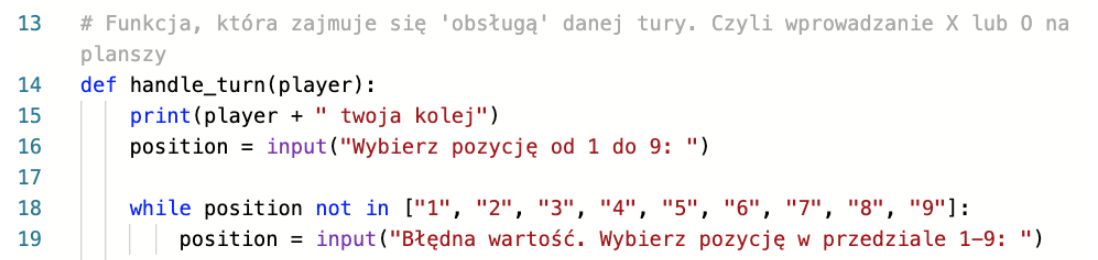
Musimy również zabezpieczyć naszą grę przed wybraniem przez użytkownika innej pozycji lub wpisaniem innego znaku niż te które mamy dostępne na planszy (od 1 do 9). Możemy w tym wypadku skorzystać z pętli, która będzie sprawdzać. czy użytkownik podał jedną z poprawnych wartości.

W naszym kodzie umieść wyrażenie while position not in ["1", "2", "3", "4", "5", "6", "7", "8", "9"]: , które w prostym tłumaczeniu oznacza, że jeżeli pozycja nie jest jednym ze znaków określonych w nawiasie kwadratowym to wywoła ono odpowiednie działanie.

Takim działaniem będzie wyświetlenie tekstu “Błędna wartość. Wybierz pozycję od 1 do 9:” i znowu próba przypisania tej wartości do zmiennej position , poprzez skorzystanie z funkcji input.

Wewnątrz naszego wyrażenia while position not in ["1", "2", "3", "4", "5", "6", "7", "8", "9"]: umieść kod: position = input("Błędna wartość. Wybierz pozycję w przedziale 1-9: ")

Powinno to wyglądać w całości następująco:



Nasza wartość w position jest aktualnie stringiem (czyli ciągiem znaków). Musimy ją zmienić na wartość liczbową, przyjmiemy do tego liczby całkowite, czyli skorzystamy z funkcji int.

Wystarczy, że w następnej linijce naszego kodu umieścisz kolejną zmienną:



Musimy również napisać skrypt, który będzie odpowiadał za podanie informacji, że pozycja wybrana przez gracza została już uzupełniona i nie można jej wybrać. Żeby to zrobić skorzystamy z warunku if.

Wystarczy, że wewnątrz funkcji handle\_turn(player) umieścisz warunek sprawdzający, czy wybrana pozycja na planszy nie jest myślnikiem i w momencie, w którym ten warunek zostanie spełniony wyświetlisz informację tekstową: “To miejsce na planszy jest już zajęte. Możesz do tego wykorzystać jeden z dostępnych w pythonie operatorów”.

W poprzednich lekcjach korzystaliśmy z operatora równości “==”. Jeżeli chcemy użyć operacji przeciwnej, która będzie oznaczała “nie równa się” to możesz użyć operatora “!=”.

Zrób to w następujący sposób:



Dodajmy jeszcze na końcu naszej funkcji handle\_turn(player) zmienną board[position] = player, oraz ponowne wyświetlenie naszej planszy poprzez wywołanie funkcji display\_board().

Cały Twój kod po tej lekcji powinien wyglądać następująco:



To już wszystko w tej lekcji. W następnej części naszego mini szkolenia, zajmiemy się stworzeniem skryptu, który sprawdzi czy gra została zakończona.