

Wszystkie materiały udostępnione w opracowaniu przeznaczone są wyłącznie dla uczniów II Liceum Ogólnokształcącego im. Mikołaja Kopernika w Bydgoszczy. Prezentowany materiał został przygotowany na potrzeby zajęć lekcyjnych z informatyki i stanowi jedynie dodatek do treści, które uczeń powinien przyswoić podczas lekcji oraz z podręcznika.

**Opracowanie:**  
**Dariusz Bagnucki**

**KALKULATOR WALUTOWY W OPARCIU O TABELĘ KURSÓW NBP**

Wartość do przeliczenia: 2 000 zł  
 Wniosek: EUR  
 Wynosi: 463,32 zł  
 pozwoli nam na zakup

**BYDGOSKA AGENCJA KOSMICZNA**  
 (oddział terenowy)  
 85-000 Bydgoszcz, ul. Gwiedźna 100  
 tel. 52 999 666 999

Bydgoszcz, 1 listopada 2023  
 Szanowna Pani  
 Krystyna Borys  
 ul. Świerzeńska 49/30  
 35-731 Nowy Dwór

**ZAGROŻENIA W S**

Gdy mówimy o zagrożeniu w sieci od fani kodów i zabezpieczeń do naszego, domowego komputera. Czy tak jest faktycznie? Otóż prawdziwi hakerzy nie są prywatnych, no chyba że o dużym zasięgu działania. Uwagę zawodowych hakerów: giełdy, serwery rządowe. Są to specjaliści o ogromnej sławy, albo dla pieniędzy, albo

**II Liceum Ogólnokształcące im. Mikołaja Kopernika w Bydgoszczy**

## PROGRAMOWANIE – Język Python. Operacje na tekstach.

Cel lekcji:

Celem lekcji jest pogłębianie wiedzy z zakresu programowania w języku PYTHON.

Podczas wykonywania ćwiczeń zapoznamy się z podstawowymi operacjami na listach danych oraz na tekstach.

Rozpoczynamy cykl ćwiczeń z programowania realizowany w oparciu o język Python.

Na dzisiejszych zajęciach będziemy korzystali z podstawowych poleceń języka, które przypomnieliśmy sobie na poprzedniej lekcji.

Podczas wykonywania ćwiczeń możemy korzystać z omówionego materiału, który znajduje się na stronie:

## [Powtórka materiału – Język programowania PYTHON.](#)

W celu realizacji zajęć musimy zlokalizować na swoim komputerze katalog, w którym PyCharm zapisuje i przechowuje pisane przez nas programy oraz przechowuje związane z nimi dane.

Wszystko, co tworzymy zapisuje się zazwyczaj w domyślnej lokalizacji na dysku, w poniższym katalogu:

C:\Users\Roboczy\PycharmProjects\...

Element ścieżki, katalog **Roboczy** jest **Nazwą Użytkownika**, w związku z czym na Waszych komputerach jego nazwa będzie inna!!!

Proszę odszukać wskazaną ścieżkę na swoim komputerze, gdyż będziemy wgrywali tam wskazane poniżej pliki.

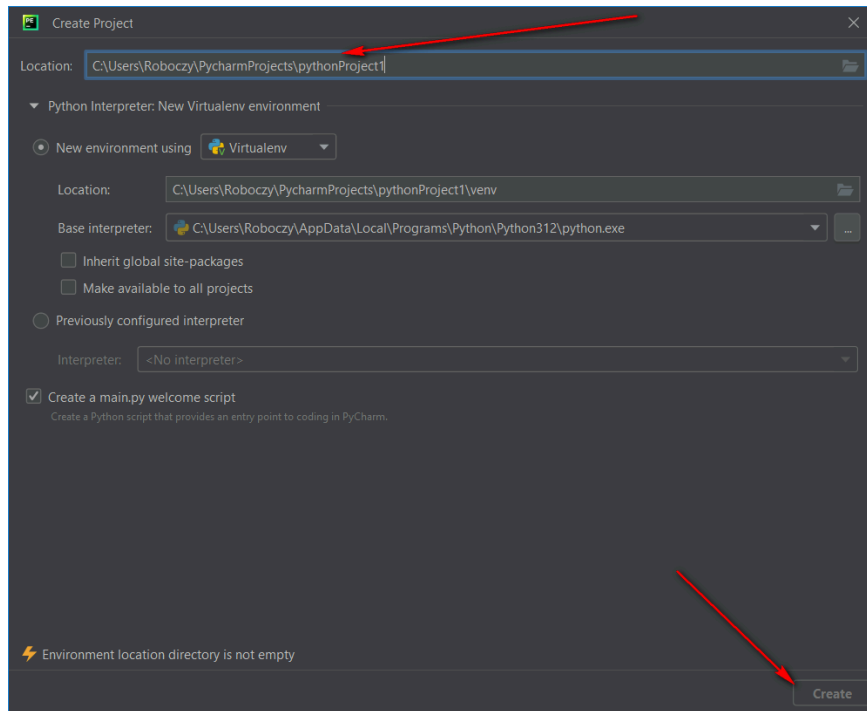
Następnie uruchamiamy program PyCharm.

Na górnej listwie wybieramy opcję **File** i z listy, która się pojawiła wybieramy **New Project**.

Na ekranie pojawi się nam okno z sugerowaną nazwą nowego projektu (w tym przypadku jest to **pythonproject1**). W miejsce **pythonproject1** **wpisujemy swoje nazwisko oraz imię oddzielone znakiem podkreślnika (podłogi)**.

**Przykładowo Bagnucki\_Dariusz.**

**W nazwie NIE UŻYWAMY POLSKICH LITER!!!**

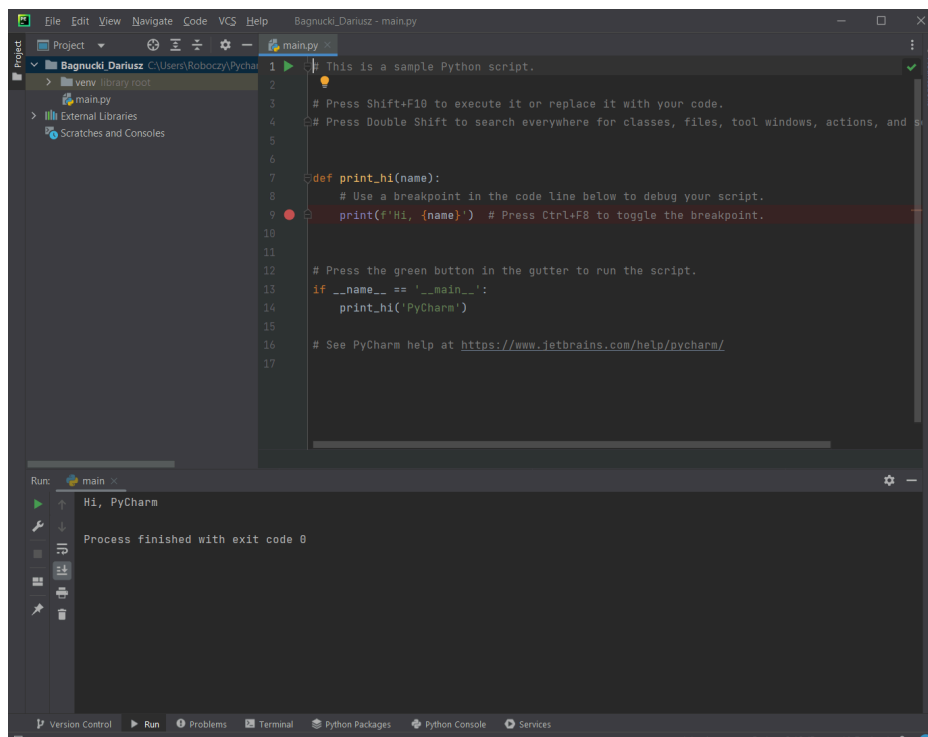


Następnie wciskamy przycisk **Create**.

Określamy, czy projekt ma otworzyć się w nowym oknie, czy w aktualnym.

Wybieramy okno aktualne.

Na ekranie zobaczymy następujący widok.



Od tej chwili wszystkie napisane przez nas programy będą automatycznie umieszczane w katalogu:

C:\Users\Roboczy\PycharmProjects\Bagnucki\_Dariusz

Gdzie jak już wspomniałem Roboczy określa nazwę użytkownika komputera, a Bagnucki\_Dariusz jest jednocześnie nazwą naszego projektu i nazwą domyślnego katalogu, w którym znajdą się dane oraz pisane programy.

Ponownie z listwy górnej wybieramy opcję **File** i z listy menu wybieramy **New**.

Wybieramy z listy opcję **File** i w oknie, które się pojawiło wpisujemy nazwę programu, który będziemy pisali. W tym przypadku wpisujemy **Zakupy.py**, pamiętając o rozszerzeniu **.py**

**Mamy już przygotowane środowisko pracy i możemy zabrać się do realizacji zadania.**

Aby ułatwić realizację ćwiczeń oraz ze względu na to, że instrukcje wejścia/wyjścia dla zapisu i odczytu danych będą omawiane podczas kolejnych lekcji skorzystamy z gotowego fragmentu kodu oraz pliku zawierającego dane do obróbki.

Aby pobrać niezbędne pliki otwieramy dysk Google przypisany do dzisiejszych zajęć.

**[DYSK GOOGLE](#)**

Pobieramy z niego plik danych o nazwie **Dane.txt** oraz fragment kodu o nazwie **Wzor.py**

**UWAGA!!!**

Kopiujemy zawartość pliku Wzor.py do naszego programu Zakupy.py, a plik Dane.txt przenosimy do katalogu domyślnego naszego projektu (jego nazwa to Wasze nazwisko oraz imię)

## ĆWICZENIE 1

Waszym zadaniem jest sprawdzenie, czy na liście zakupów zapisanych w pliku Dane.txt nie zostały powtórzone pozycje. Należy zabezpieczyć się przed kupieniem danego produktu wielokrotnie.

Jeżeli znajdziemy takie pozycje, należy je zastąpić szeregiem znaków -----.

Zaczynamy od przygotowania algorytmu.

W celu jego stworzenia posługujemy się poznanym wcześniej programem DRAW.IO

Przypominam, wersja OnLine tego programu znajduje się na stronie:

### Draw.IO

Wszyscy doskonale poznaliśmy działanie tego pakietu tworząc w poprzedniej klasie algorytm śpiewania piosenki STO LAT 😊

Dla zaoszczędzenia czasu skupimy się na tworzeniu części algorytmu związanego z głównym zadaniem ćwiczenia, czyli z wyszukaniem oraz eliminacją powtórzeń zapisu.

Procedury wejścia/wyjścia przedstawimy w postaci bloków, jak na poniższym rysunku:



W ćwiczeniu posługujemy się wyłącznie elementami języka, które omawiane były wcześniej na lekcjach. Nie korzystamy z gotowych funkcji, procedur ani bibliotek.

Ćwiczenie, w którym zastosowane zostaną takie rozwiązania nie zostanie zaliczone!!!

Na dysku Google przypisanym do Waszej klasy tworzymy katalog zgodnie z przyjętym nazewnictwem i wgrywamy do niego gotowy, sprawdzony i poprawnie działający program, plik Dane.txt, plik Zakupy.txt oraz plik zawierający Wasz schemat blokowy.

## ZAWARTOŚĆ PLIKU WZOR.PY

### # OBSZAR PROCEDUR WYWOŁYWANYCH W PROGRAMU GŁÓWNEGO

"""

*Procedura otwiera plik zawierający nasze dane, czyli nieuporządkowaną listę zakupów. Każdą odczytaną linię dodaje do utworzonej listy, jako jej kolejny element.*

*Powstała lista wygląda w sposób następujący:*

*['Chleb', 'Masło', 'Jajka', 'Marchew', 'Chleb', 'Seler', 'Cukinia', 'jajka', 'Majonez', 'Ser', 'Szynka', 'Kalarepa', 'Ser', 'Ser', 'Chleb']*

*Jak widać mamy na liście pozycje, które się powtarzają!!!*

*Program musi je usunąć, posortować i zapisać w pliku Zakupy.txt*

"""

```
def wczytaj_liste_zakupow(nazwa_pliku):
```

```
    # Odczytanie zawartości pliku i zwrócenie listy zakupów
```

```
    with open(nazwa_pliku, 'r', encoding='utf-8') as plik:
```

```
        lista = plik.read().splitlines()
```

```
    return lista
```

"""

*Procedura tworzy plik wynikowy, w którym zapisuje gotową listę zakupów bez powtarzających się pozycji.*

*Poza nazwami artykułów z pliku Dane.txt do pliku wynikowego dodawany jest numer każdego wiersza.*

*Dzięki temu wiemy, ile produktów mamy kupić.*

"""

```
def zapisz_liste_do_pliku(lista, nazwa_pliku):
```

```
    # Zapisanie listy zakupów do nowego pliku
```

```
    with open(nazwa_pliku, 'w', encoding='utf-8') as plik:
```

```
        licznik=1
```

```
        for pozycja in lista:
```

```
plik.write(str(licznik) + ' - ' + pozycja + '\n')
licznik +=1
```

```
"""
```

To jest procedura, którą należy dopisać samodzielnie. Ma ona za zadanie usunięcie powtórzeń z danych początkowych.

```
"""
```

```
def usun_powtorzenia(lista):
```

```
    # Nie używany wbudowanej funkcji języka, usuwającej powtarzające się elementy,
    #która wygląda w ten sposób:
```

```
    # lista_bez_powtorzen = list(dict.fromkeys(lista)).
```

```
    # Duplikaty usuwamy tworząc pętlę i usuwając pojedynczo elementy listy.
```

```
    # W TYM MIEJSCU WPISUJEMY BRAKUJĄCY KOD, KTÓRY BĘDZIE ZAMIENIAŁ
```

```
    # POWTARZAJĄCE SIĘ PRODUKTY NA ZNAKI -----
```

```
lista_bez_powtorzen = lista
```

```
return lista_bez_powtorzen
```

```
# OBSZAR PROGRAMU GŁÓWNEGO
```

```
nazwa_pliku_wejsciowego = 'Dane.txt'
```

```
nazwa_pliku_wyjsciowego = 'Zakupy.txt'
```

```
# Odczytanie listy zakupów z pliku
```

```
lista_zakupow = wczytaj_liste_zakupow(nazwa_pliku_wejsciowego)
```

```
# Usunięcie powtarzających się pozycji
```

```
lista_zakupow_bez_powtorzen = usun_powtorzenia(lista_zakupow)
```

```
# Zapisanie zaktualizowanej listy do nowego pliku
```

```
zapisz_liste_do_pliku(lista_zakupow_bez_powtorzen, nazwa_pliku_wyjsciowego)
```

```
print(f"Lista zakupów została zapisana w pliku {nazwa_pliku_wyjsciowego} bez powtórzeń.")
```

W wyniku działania programu zapewne otrzymaliście listę wyglądającą w następujący sposób:

1 - Chleb

2 - Masło

3 - Jajka

4 - Marchew

5 - -----

6 - Seler

7 - Cukinia

8 - jajka

9 - Majonez

10 - Ser

11 - Szynka

12 - Kalarepa

13 - -----

14 - -----

15 - -----

16 - -----

Taka postać wystarczy na ocenę dostateczną ... Aby uzyskać ocenę bardzo dobrą trzeba się odrobinę postarać 😊