

Zadanie 1

Napisz prosty program, który na podstawie podanej przez Studenta liczby zdobytych punktów, poinformuje go o rezultacie egzaminu.

Każdy Student, który zdobył powyżej 80 punktów zalicza egzamin w terminie 0

Studenci którzy otrzymali liczbę punków z przedziału 50-80, mogą poprawić jego wynik.

Studenci, którzy zdobyli poniżej 50 punktów, muszą go poprawić.

Przebieg: Piszemy program który na podstawie zmiennych i funkcji if, informuje nas o tym czy student zaliczył egzamin, czy może go poprawić albo czy musi go poprawić.

Kod:

```
zad1.py > ...
1 print("Zadanie 1")
2 print("\n")
3
4 punkty = int(input("Wprowadź zdobytą liczbę punktów."))
5
6 if punkty >= 80:
7     print("Egzamin zaliczony w terminie 0.")
8 elif punkty > 50 and punkty < 80:
9     print("Możliwa poprawa egzaminu.")
10 else:
11     print("Musisz poprawić egzamin.")
```

Wynik kodu:

```
Zadanie 1

Wprowadź zdobytą liczbę punktów.68
Możliwa poprawa egzaminu.
```

Analiza kodu:

Użytkownik wpisuje z klawiatury ile punktów otrzymał student i przetrzymuje to w zmiennej punkty, następnie za pomocą funkcji if sprawdza czy punkty są większe lub równe 80 gdy są drukuje że egzamin zaliczony gdy nie to sprawdza czy punkty mieszczą się w granicy więcej od 50 a mniej od 80 wtedy drukuje że poprawa jest możliwa, w przypadku gdy dalej się punkty w tym zakresie nie mieszczą drukuje że trzeba poprawić egzamin.

Zadanie 2

Napisz program porządkowania trzech liczb x, y i z. Od najmniejszej do największej, bez użycia wbudowanych funkcji

Przebieg: Piszymy kod za pomocą rozbudowania funkcji if na wszystkie możliwości żeby kod nam uporządkował wprowadzone przez nas liczby od najmniejszej do największej.

Kod:

```
 zad2.py > ...
1  print("Zadanie 2")
2  print("\n")
3
4  x = int(input("Wprowadź liczbę x: "))
5  y = int(input("Wprowadź liczbę y: "))
6  z = int(input("Wprowadź liczbę z: "))
7
8  if x > y and x > z:
9      if y > z:
10         print(f"Porządek liczb od najmniejszej do największej: {z}, {y}, {x}")
11     elif z > y:
12         print(f"Porządek liczb od najmniejszej do największej: {y}, {z}, {x}")
13     elif y > x and y > z:
14         if x > z:
15             print(f"Porządek liczb od najmniejszej do największej: {z}, {x}, {y}")
16         elif z > x:
17             print(f"Porządek liczb od najmniejszej do największej: {x}, {z}, {y}")
18     elif z > x and z > y:
19         if x > y:
20             print(f"Porządek liczb od najmniejszej do największej: {y}, {x}, {z}")
21         elif y > x:
22             print(f"Porządek liczb od najmniejszej do największej: {x}, {y}, {z}")
23
24
25
```

Wynik kodu:

```
Zadanie 2

Wprowadź liczbę x: 33
Wprowadź liczbę y: 1
Wprowadź liczbę z: 99
Porządek liczb od najmniejszej do największej: 1, 33, 99
```

Analiza kodu:

Po wprowadzeniu przez użytkownika 3 różnych liczb kod sprawdza za pomocą funkcji if zależności między liczbami w każdej możliwej kombinacji, następnie drukuje porządek od najmniejszej do największej.

Zadanie 3

Zmienna Nazwa_pliku przechowującej jego nazwę.

Sprawdź, czy plik o podanej nazwie jest z rozszerzeniem '.xlsx'.

Nazwa_pliku= 'Raport_maj.xlsx'

Wydrukuj do konsoli 'Tak' jeśli to prawda, przeciwnie 'Nie'.

Podpowiedź:

Skorzystaj z metody endswith (jest to metodą wbudowaną dla obiektów typu str w Pythonie), sprawdza ona, czy ciąg znaków (Nazwa_pliku) kończy się podanym sufiksem (w tym przypadku '.xlsx').

Przebieg: Sprawdzamy za pomocą metody endswith czy przez nas podany plik kończy się rozszerzeniem „.xlsx” i zależnie czy tak czy nie jest to drukowanie do konsoli.

Kod:

```
zad3.py > ...
1  print("Zadanie 3")
2  print("\n")
3
4  Nazwa_pliku = "Raport_maj.xlsx"
5
6  if Nazwa_pliku.endswith('.xlsx'):
7      print("Tak")
8  else:
9      print("Nie")
```

Wynik kodu:

```
Zadanie 3

Tak
```

Analiza kodu:

Dodajemy zmienną z naszą nazwą pliku, następnie za pomocą funkcji if sprawdzamy przy pomocy metody endswith czy nasz plik kończy się podanym przez nas rozszerzeniem w tym przypadku „.xlsx”. Następnie kod drukuje wynik „Tak” lub „Nie” zależnie od spełnienia warunku.

Zadanie 4

Jesteś menagierem drużyny piłkarskiej i chcesz obliczyć łączny wynik drużyny na podstawie liczby strzelonych przez nią bramek i ewentualnie zdobytych dodatkowych punktów. Napisz program, który dokona stosownych kalkulacji po wprowadzeniu liczby goli zdobytych przez drużynę. Utworzono są dwie zmienne gol i bonus, gdzie gol to liczba całkowita reprezentująca liczbę bramek zdobytych przez drużynę, a bonus to liczba całkowita reprezentująca wszelkie możliwe punkty bonusowe dla drużyny.

Następnie użyj instrukcji warunkowej do obliczenia całkowitego wyniku zespołu zgodnie z następującymi zasadami:

- każda zdobytą bramka to 10 punktów,
- jeśli drużyna zdobędzie więcej niż 5 bramek, zdobywa dodatkowe 5 punktów bonusowych,
- jeśli drużyna zdobędzie więcej niż 10 bramek, zdobywa dodatkowe 10 punktów bonusowych

a) Po zdobyciu 5 goli drużyna otrzymuje 5 punktów bonusowych. Jeśli drużyna zdobędzie więcej niż 10 goli, to otrzyma za nie 10 punktów bonusowych dodatkowo

Oblicz całkowity wynik drużyny, dodając punkty zdobyte ze zdobytych bramek i wszelkie stosowne punkty bonusowe. Wynik wydrukuj do konsoli.

b) Punkty bonusowe po przekroczeniu 5 i 10 punktów są sumowane, tzn. po przekroczeniu więcej niż 10 bramek drużyna zdobywa obydwa bonusy.

Oblicz całkowity wynik drużyny, dodając punkty zdobyte ze zdobytych bramek i wszelkie stosowne punkty bonusowe. Wynik wydrukuj do konsoli.

Przebieg: Za pomocą instrukcji warunkowej obliczamy całkowity wynik meczu przy określonych zależnościach.

Kod:

```
zad4.py > ...
1  print("Zadanie 4")
2  print("\n")
3
4  print("A)")
5  gole = int(input("Wprowadź liczbę goli strzelonych przez drużynę: "))
6  bonus = 0
7
8  if gole > 10:
9    |   bonus = 10
10 elif gole > 5:
11   |   bonus = 5
12
13 wynik = gole + bonus
14
15 print(f"Całkowity wynik wynosi: {wynik}")
16
17 print("\nB)")
18 gole2 = int(input("Wprowadź liczbę goli strzelonych przez drużynę: "))
19 bonus2 = 0
20
21 if gole2 > 5:
22   |   bonus2 = 5
23   |   if gole2 > 10:
24     |     |   bonus_calowity = bonus2 + 10
25   |   else:
26     |     |   bonus_calowity = bonus2
27
28 wynik2 = gole2 + bonus_calowity
29 print(f"Całkowity wynik wynosi: {wynik2}")
```

Wynik kodu:

Zadanie 4

A)

Wprowadź liczbę goli strzelonych przez drużynę: 25
Całkowity wynik wynosi: 35

B)

Wprowadź liczbę goli strzelonych przez drużynę: 30
Całkowity wynik wynosi: 45

Analiza kodu:

Program liczy wynik drużyny na podstawie liczby strzelonych goli. Jeśli drużyna zdobędzie więcej goli, dostaje dodatkowe punkty. Dodajemy dwie zmienne, gole oraz bonus. Następnie program za pomocą funkcji zlicza ile punktów bonusowych zdobyła a końcowo liczy całkowity wynik. W podpunkcjach A) dodawaliśmy tylko 5 albo 10 punktów zależnie czy drużyna zdobyła więcej jak 5 lub 10 goli. W podpunkcie B) jeśli drużyna zdobyła więcej niż 10 goli oba bonusy zostały dodawane.

Zadanie 5

a)

Odczytaj podany plik notowania_gieldowe.txt zawierający dane dotyczące notowań kilku spółek.

Wydrukuj każdą linię do konsoli.

b)

Dopisz do pliku notowania_gieldowe.txt, w kolejnej linii dane dotyczące nowej spółki: ALR, 113.

Wydrukuj каждą linię ponownie do konsoli.

Przebieg: Program odczytuje plik, następnie drukuje każdą linijkę kodu. W drugim podpunkcie dopisujemy do pliku w kolejnej linicje nazwę nowej spółki i również drukujemy każdą linijkę kodu.

Kod:

```
zad5.py > ...
1  print("Zadanie 5")
2  print("\n")
3
4  print("A")
5
6  zawartosc_pliku = open("notowania_gieldowe.txt", "r").read()
7  print(zawartosc_pliku)
8
9  print("\nB")
10
11 with open("notowania_gieldowe.txt", "a") as file:
12     file.write("\nALR, 113")
13
14 with open("notowania_gieldowe.txt", "r") as file:
15     zawartosc_po_dopisaniu = file.read()
16     print(zawartosc_po_dopisaniu)
```

Wynik kodu:

Zadanie 5

A)

KGHM, 123
Tauron, 150
Orange, 45
PGE, 24
PKN Orlen, 70
PKO BP, 56

B)

KGHM, 123
Tauron, 150
Orange, 45
PGE, 24
PKN Orlen, 70
PKO BP, 56
ALR, 113

Analiza kodu: Za pomocą metod odczytu, zapisu pliku, dodajemy zmienna w której nasz plik jest odczytywany lub zapisywany, następnie drukujemy ten wynik w konsoli.

Zadanie 6

Napisz skrypt w Pythonie, który sprawdza, czy litera wprowadzona przez użytkownika jest duża czy mała

Przebieg:

Kod:

```
zad6.py > ...
1  print("Zadanie 6")
2  print("\n")
3
4  litera = str(input("Wprowadź literę: "))
5
6  if litera.islower():
7      print("To jest litera mała.")
8  elif litera.isupper():
9      print("To jest litera wielka.")
10
```

Wynik kodu:

```
Zadanie 6

Wprowadź literę: D
To jest litera wielka.
```

Analiza kodu:

Do zmiennej litera wpisujemy z klawiatury literę, która jest następnie sprawdzana przy pomocy funkcji if oraz metod islower oraz isupper, gdy litera jest mała drukuje nam taką informację, gdy wielka analogicznie drukuje też taką informację.

Zadanie 7

Podana jest poniższa zmienna przechowująca ciąg znaków - hasło:

Hasło = 'pk47!jy0893'

Sprawdź, czy podane hasło ma wymaganą długość 11 znaków oraz zwiera znak specjalny '!'. Jeżeli tak, wydrukuj do konsoli „Hasło jest poprawne”, w przeciwnym razie „Hasło jest nie poprawne”.

Przebieg: Przypisujemy do zmiennej 'hasło' podane hasło następnie sprawdzamy czy hasło spełnia przez nas podane warunki, jeśli tak drukujemy że spełnia, jeśli nie drukujemy że nie jest ono poprawne.

Kod:

```
zad7.py > ...
1  print("Zadanie 7")
2  print("\n")
3
4  Hasło = 'pk47!jy0893'
5
6  if len(Hasło) >= 11 and "!" in Hasło:
7      print("Hasło jest poprawne.")
8  else:
9      print("Hasło jest niepoprawne.")
```

Wynik kodu:

```
Zadanie 7

Hasło jest poprawne.
```

Analiza kodu:

Przypisujemy hasło do zmiennej, następnie za pomocą funkcji if sprawdzamy zależności czy hasło ma długość większą lub równą 11 przy pomocy funkcji len() oraz za pomocą sprawdzenia czy ma też w sobie „!”. Jeśli spełnia warunki funkcji program drukuje nam że hasło jest poprawne w przeciwnym wypadku drukuje nam że hasło nie jest poprawne.

Zadanie 8

Stwórz program, który wykorzystując operator wycinania z podanego ciąg znaków (zmienna text) wyodrębnia:

pierwsze trzy znaki

ostatnie dwa znaki

text = 'Studiuje-Informatykę'

Wynik wydrukuj do konsoli.

Przebieg: Tworzymy program i z tekstu w zmiennej wycinamy 3 pierwsze znaki oraz dwa ostatnie znaki.

Kod:

```
zad8.py > ...
1  print("Zadanie 8")
2  print("\n")
3
4  text = 'Studiuje-Informatykę'
5
6  trzyznakipierw = text[0:3]
7  print(trzyznakipierw)
8
9  ostatniedwa = text[-2:]
10 print(ostatniedwa)
11
```

Wynik kodu:

Zadanie 8

Stu
kę

Analiza kodu: Korzystając z operatora wycinania, do nowej zmiennej przypisujemy wycięty przez nas skrawek tekstu następnie go drukujemy.

Zadanie 9

Napisz skrypt zmieniający wszystkie duże litery małe i na odwrót.
Podpowiedź: skorzystaj z metody swapcase().

Przebieg: Zmieniamy wszystkie duże litery na małe i odwrotnie.

Kod:

```
zad9.py > ...
1  print("Zadanie 9")
2  print("\n")
3
4  tekst = str(input("Wprowadź tekst: "))
5
6  tekst_odwrocony = tekst.swapcase()
7  print(tekst_odwrocony)
```

Wynik kodu:

Zadanie 9

Wprowadź tekst: LoSOwy Tekst

10soWY tEKSt

Analiza kodu:

Do zmiennej wpisujemy z klawiatury przez nas podany tekst, następnie do nowej zmiennej przypisujemy ten tekst korzystając z metody swapcase(). Na sam koniec go drukujemy.