

```

class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("Wpisz swój PESEL");
        // zapisujemy w zmiennej o nazwie pesel, tekst który wpisał Użytkownik
        string pesel = Console.ReadLine();
        // sprawdzamy czy Użytkownik wpisał cokolwiek, a nie tylko nacisnął Enter
        if (pesel == "")
        {
            // to oznacza, że nic nie wpisał, tylko nacisnął Enter
            // ustawiamy kolor czcionki na czerwony
            Console.ForegroundColor = ConsoleColor.Red;
            // wyświetlamy komunikat
            Console.WriteLine("Nie wpisałeś nic => Koniec programu");
            // Aby wyświetlić ponownie komunikat żeby wpisał PESEL bez zakończenia programu
            // (czyli jeżeli chcielibyśmy czekać do skutku aż w końcu łaskawie Użytkownik
            // wpisze cokolwiek w konsoli) => musielibyśmy użyć pętli while ... której jeszcze
            // na tym etapie nauki nie znamy. Dlatego kończymy program, jeżeli Użytkownik nie wpisze.
        }
        else
        {
            // to oznacza, że Użytkownik wpisał cokolwiek
            // teraz musimy zatem sprawdzić czy to co wpisał wogóle przypomina PESEL
            // przede wszystkim PESEL musi się składać z 11 cyfr
            // najpierw sprawdzimy czy to co wpisał Użytkownik ma 11 znaków
            if (pesel.Length != 11)
            {
                // znaczy że długość tekstu który wpisał jest innej długości niż 11 znaków
                // metoda Length liczy również spacje
                // czyli z pewnością nie wpisał prawidłowego PESELU
                // ustawiamy kolor czcionki na czerwony
                Console.ForegroundColor = ConsoleColor.Red;
                // wyświetlamy komunikat
                Console.WriteLine("PESEL musi się składać z 11 cyfr => Koniec programu");
            }
            else
            {
                // znaczy, że długość jest ok => 11 znaków
                // możemy sprawdzać dalej
                // teraz sprawdzamy czy wpisał same cyfry
                // mamy kilka możliwości sprawdzenia tego
                // moglibyśmy np: użyć pętli i sprawdzić czy każdy znak we wpisanym tekście jest cyfrą
                // ja to zrobię inaczej:
                // spróbuję zamienić sobie tekst, który wpisał na liczbę typu long
                // wykorzystam do tego gotową już metodę TryParse (daje ją Nam Microsoft w prezencie)
                // metoda long.TryParse zwraca typ bool (prawda lub fałsz)
                // jeżeli uda się tej metodzie zamienić tekst na liczbę typu long wówczas zwróci wartość true
                // oraz dodatkowo (jako parametr zewnętrznie zwracany => o tym nie będziemy narazie mówić)
                // właśnie tę liczbę typu long którą sobie zrobił z tego tekstu
                bool czyWpisałSameCyfry = long.TryParse(pesel, out long liczba);
                // np: gdy ktoś wpisze tekst: "12345678932" metoda ta zwróci true (bo jest w stanie
            }
        }
    }
}

```

```

// zamienić ten tekst na liczbę)
// a tą liczbą będzie liczba: 12345678932
// metoda nie akceptuje spacji czyli jeżeli ktoś wpisze np: "32232 34543" metoda zwróci false
// liczba ta nie jest Nam dalej do niczego potrzebna (po prostu tak jest zbudowana
// ta metoda, że dodatkowo zwraca tę liczbę)
// nas interesuje jedynie wynik którą zwraca metoda (czyli true albo false)
// więc sprawdźmy zatem:
if (czyWpisałSameCyfry)
{
    // skoro jesteśmy w tym miejscu oznacza to z pewnością, że Użytkownik
    // pięknie wpisał 11 samych cyfr
    // możemy zatem sprawdzić czy jest to prawidłowy numer PESEL
    // służy do tego metoda którą napisałem poniżej o nazwie CzyPrawidlowyPESEL
    // zwraca ona typ bool czyli true albo false
    // przekazujemy do niej jako parametr tekst wpisany przez Użytkownika
    // pamiętajmy, że wiemy już że wpisał 11 samych cyfr.
    // sprawdzamy
    if (CzyPrawidlowyPESEL(pesel))
    {
        // oznacza to, że PESEL jest prawidłowy
        // wyświetlamy komunikat => tym razem na zielono
        Console.ForegroundColor = ConsoleColor.Green;
        Console.WriteLine("Prawidłowy PESEL");

    }
    else
    {
        // oznacza to, że PESEL niestety nie jest prawidłowy
        // ustawiamy kolor czcionki na czerwony
        Console.ForegroundColor = ConsoleColor.Red;
        // wyświetlamy komunikat
        Console.WriteLine("Nieprawidłowy PESEL");

    }
}
else
{
    // skoro jesteśmy w tym miejscu oznacza to że, co prawda Uzytkownik wpisał 11 znaków,
    // ale niestety nie wszystkie z nich są cyframi
    // mógł np wpisać: "xcdretyikw", albo "345889 908"
    // (spacja to nie cyfra), albo "rt5689d787w"
    // coś takiego napewno nie jest pesel
    // więc:
    // ustawiamy kolor czcionki na czerwony
    Console.ForegroundColor = ConsoleColor.Red;
    // wyświetlamy komunikat
    Console.WriteLine("PESEL musi się składać tylko z 11 cyfr => Koniec programu");
}
}
}

Console.ReadLine();
}

```

```

// metoda sprawdzająca poprawność PESEL
static bool CzyPrawidlowyPESEL(string pesel)
{
    // wiemy już, że to co wpisał użytkownik składa się z 11 znaków
    // i co więcej, są to same cyfry
    // tego kodu możecie nie rozumieć => nie ma problemu
    // po prostu za mało jeszcze wiecie
    // tak tylko informacyjnie:
    // to jest algorytm sprawdzania poprawności numeru PESEL
    // PESEL składa się z 11 cyfr
    // pierwsze 10 to informacje o posiadaczu (min. data urodzenia, płeć)
    // ostatnia cyfra to tzw: cyfra kontrolna
    // generalnie algorytm polega min. sprawdzeniu poprawności tej cyfry kontrolnej

    int[] wagi = { 1, 3, 7, 9, 1, 3, 7, 9, 1, 3 };
    int suma = 0;
    int cyfraKontrolna = int.Parse(pesel.Substring(10, 1));

    for (int idx = 0; idx < wagi.Length ; idx++)
    {
        suma += int.Parse(pesel.Substring(idx, 1)) * wagi[idx];
    }

    suma = suma % 10;

    if ((10 - suma) % 10 == cyfraKontrolna)
    {
        if (int.Parse(pesel.Substring(2, 2)) > 32)
            return false;
        else if (pesel == "2222222222")
            return false;
        else
            return true;
    }
    else
    {
        return false;
    }
}
}

```