

EGZAMIN ZAWODOWY – Arkusz próbny

Kwalifikacja: INF.04 – Projektowanie, programowanie i testowanie aplikacji

Część praktyczna

Czas trwania: 180 minut

Numer zadania: 01

Instrukcja dla zdającego

1. Na pierwszej stronie arkusza wpisz swój numer PESEL.
 2. W folderze roboczym utwórz katalog o nazwie *numer_zdajacego*.
 3. W katalogu utwórz podfoldery: **desktop**, **konsolowa**, **dokumentacja**.
 4. Po wykonaniu zadań zapisz kod, dokumentację oraz zrzuty ekranu zgodnie z opisem.
 5. Pracuj samodzielnie i przestrzegaj zasad bezpieczeństwa.
-

ZADANIE EGZAMINACYJNE

Masz wykonać **aplikację konsolową** oraz **aplikację desktopową**, a następnie przygotować **dokumentację** zgodnie z częścią III zadania.

Część I – Aplikacja konsolowa

Napisz program implementujący **klasę bazową** oraz **dwie klasy potomne**, służący do obsługi **samochodów**.

Opis wymagań:

1. Klasa bazowa – Pojazd

- Udostępnia jedną metodę:
wyswietlKomunikat(tresc) – wyświetla komunikat przekazany w parametrze.
Nic nie zwraca.
-

2. Klasa potomna – SamochodOsobowy

- Posiada prywatne pole **maksymalnaPredkosc**, typu całkowitego, domyślnie **0**.

- Udostępnia metodę:
ustawPredkosc(v) – jeżeli *v* mieści się w zakresie **1–250**, ustawia pole i zwraca jego wartość; w przeciwnym razie ustawia 0 i zwraca 0.
 - Program wyświetla komunikaty zgodne z działaniem (przykład w widoku programu).
-

3. Klasa potomna – SamochodCiezarowy

- Posiada prywatne pole **ladownosc**, typu całkowitego, domyślnie **0**.
 - Udostępnia metodę:
ustawLadownosc(kg) – jeżeli wartość mieści się w zakresie **500–30000**, ustawia pole; inaczej ustawia **0**.
 - Udostępnia metodę **zaladuj()**, która:
 - jeśli ładowność > 0:
wywołuje metodę bazową z komunikatem: „Samochód ciężarowy załadowano”
 - jeśli ładowność = 0 – nic nie robi.
-

4. Program główny

Program powinien:

1. Utworzyć obiekty *SamochodOsobowy* i *SamochodCiezarowy*.
 2. Przetestować wprowadzanie poprawnych i niepoprawnych wartości (prędkości, ładowności).
 3. Wyświetlać komunikaty zrozumiałe dla użytkownika.
 4. Być napisany w jednym z języków dostępnych na stanowisku:
C#, C++, Java, Python.
-

Wygenerowana grafika – przykład działania aplikacji konsolowej

(grafika została wygenerowana automatycznie zgodnie z Twoją prośbą)

attachment:turn2image0

Część II – Aplikacja desktopowa

Utwórz aplikację desktopową (WinForms / WPF / MAUI Desktop / GTK / Qt – zgodnie z dostępnością na stanowisku).

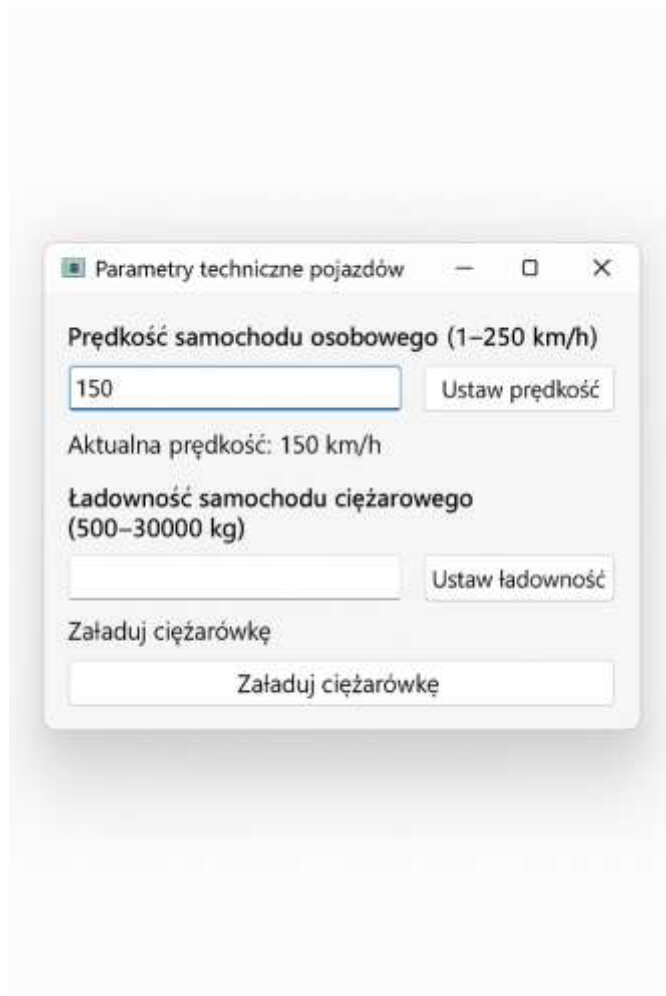
Wymagania funkcjonalne:

Interfejs powinien zawierać:

1. Nagłówek: „**Parametry techniczne pojazdów**”
2. Pole edycyjne „Prędkość samochodu osobowego (1–250 km/h)”
3. Przycisk „**Ustaw prędkość**”
4. Etykietę wynikową: „Aktualna prędkość: —”
5. Pole edycyjne „Ładowność samochodu ciężarowego (500–30000 kg)”
6. Przycisk „**Ustaw ładowność**”
7. Przycisk „**Załaduj ciężarówkę**”
8. Etykietę wynikową: „Status ciężarówki: —”

Zasady działania:

- Po naciśnięciu „Ustaw prędkość” program waliduje dane jak w aplikacji konsolowej.
- Po naciśnięciu „Ustaw ładowność” – analogicznie.
- Po naciśnięciu „Załaduj ciężarówkę”:
 - jeśli ładowność > 0 → komunikat: „Samochód ciężarowy załadowano”
 - inaczej → nic się nie zmienia



Część III – Dokumentacja

W podfolderze **dokumentacja** umieść:

1. Komentarz dokumentacyjny w kodzie

W klasie bazowej nad lub pod definicją metody dodaj dokumentację w formacie:

```
*****  
nazwa: wyswietlKomunikat  
opis: Wyświetla treść komunikatu  
parametry: tresc - tekst do wyświetlenia  
zwracany typ i opis: brak  
autor: <numer zdającego>  
*****
```

2. Zrzuty ekranu

- Aplikacja konsolowa – minimum 2 zrzuty
- Aplikacja desktopowa – minimum 3 zrzuty

Zrzuty muszą obejmować cały ekran wraz z paskiem zadań.

3. Dokument tekstowy

Plik **egzamin.docx** zawierający:

- system operacyjny
- środowiska programistyczne
- języki programowania
- kroki wykonania