

Zadanie 1

1. Tytuł Zadania

Aplikacja: Rejestracja Czasu Pracy

2. Opis Interfejsu (GUI)

W oknie aplikacji desktopowej (Windows Forms lub WPF):

- Tytuł okna musi zawierać numer PESEL zdającego, np. „Rejestracja czasu pracy – 01234567890”.
- Kontrolki:
 - 2 etykiety (Label): „Godzina rozpoczęcia” i „Godzina zakończenia”.
 - 2 pola tekstowe (TextBox) do wprowadzania godzin w formacie HH:MM (np. 08:30, 16:15).
 - Etykieta „Przerwa (minuty)” oraz TextBox do wprowadzenia liczby minut przerwy.
 - Etykieta „Stawka za godzinę (zł)” oraz TextBox na stawkę godzinową.
 - Przycisk (Button) „Oblicz”.
 - Pole tekstowe tylko do odczytu lub Label z wynikami: „Przepracowane godziny:” i „Kwota do wypłaty:”.
- Początkowo pola wynikowe są puste.
- Przycisk „Oblicz” jest aktywny od początku.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu przycisku „Oblicz” program:
 - Sprawdza, czy wszystkie pola (godzina rozpoczęcia, godzina zakończenia, przerwa, stawka) są wypełnione.
 - Sprawdza, czy godziny są w formacie HH:MM:
 - * Weryfikuje występowanie znaku „:”.
 - * Sprawdza, czy godziny są w zakresie 0–23, a minuty 0–59.
 - Sprawdza, czy przerwa i stawka są liczbami dodatnimi.
- Obliczenia:
 - Oblicza całkowity czas pracy w minutach jako różnicę między godziną zakończenia a rozpoczęcia.
 - Od całkowitego czasu odejmuje czas przerwy (minuty).
 - Wylicza przepracowany czas w godzinach (z dokładnością do dwóch miejsc po przecinku).
 - Oblicza kwotę do wypłaty:
$$kwota = \text{przepracowane godziny} \times \text{stawka godzinowa}.$$
- Jeżeli przerwa jest większa niż całkowity czas pracy:

- Wyświetlany jest komunikat (MessageBox) o błędzie.
- Obliczenia nie są wykonywane, pola wynikowe pozostają puste lub niezmienione.
- Wynik jest wyświetlany w polach wynikowych, np.:
 - „Przepracowane godziny: 7,50”
 - „Kwota do wypłaty: 150,00 zł”
- W przypadku niepoprawnych danych wejściowych (niepoprawny format czasu, puste pola, nieliczbowe wartości):
 - Wyświetlany jest MessageBox z informacją o przyczynie błędu.

4. Kryteria Oceniania

- R.3.1: Okno aplikacji ma tytuł zawierający numer PESEL zdającego.
- R.3.2: Interfejs zawiera wszystkie wymagane kontrolki i są one logicznie rozmieszczone oraz opisane.
- R.3.3: Program sprawdza, czy pola wejściowe nie są puste i czy wartości liczbowe (przerwa, stawka) są dodatnimi liczbami.
- R.3.4: Program poprawnie oblicza czas pracy na podstawie godzin rozpoczęcia i zakończenia oraz czasu przerwy.
- R.3.5: Program poprawnie oblicza kwotę do wypłaty na podstawie przepracowanych godzin i stawki godzinowej.
- R.3.6: Program wyświetla komunikat błędu (MessageBox) w przypadku niepoprawnych danych (np. zbyt duża przerwa, zły format czasu).
- R.3.7: Wyniki są prezentowane w czytelnej formie (z dwoma miejscami po przecinku, z jednostkami).

Zadanie 2

1. Tytuł Zadania

Aplikacja: Kalkulator Zamówienia Pizzy

2. Opis Interfejsu (GUI)

- Tytuł okna: „Kalkulator pizzy – PESEL zdającego”.
- Kontrolki:
 - Grupa (GroupBox) „Rozmiar pizzy” z trzema RadioButton:
 - * „Mała (20 zł)”
 - * „Średnia (30 zł)”
 - * „Duża (40 zł)”
 - Grupa (GroupBox) „Dodatki (5 zł każdy)” z kilkoma CheckBox:
 - * „Ser”
 - * „Szynka”

- * „Piecarki”
- * „Kukurydza”
- Etykieta „Rabat (%)” i TextBox na procent rabatu.
- Przycisk „Oblicz cenę”.
- Label lub TextBox (tylko do odczytu) z podsumowaniem:
 - * „Cena przed rabatem:”
 - * „Rabat:”
 - * „Cena po rabacie:”
- Domyślnie zaznaczony rozmiar: „Średnia”.
- Pole rabatu domyślnie ustawione na „0”.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Oblicz cenę”:
 - Program sprawdza:
 - * Czy wybrany jest rozmiar pizzy (jeden z RadioButton jest zaznaczony).
 - * Czy rabat jest liczbą z przedziału 0–50 (w procentach).
 - Obliczenia:
 - * $\text{Cena bazowa} = \text{cena wybranego rozmiaru}$.
 - * $\text{Cena dodatków} = \text{liczba zaznaczonych dodatków} \times 5 \text{ zł}$.
 - * $\text{Cena przed rabatem} = \text{cena bazowa} + \text{cena dodatków}$.
 - * $\text{Kwota rabatu} = \text{cena przed rabatem} \times (\text{rabat} / 100)$.
 - * $\text{Cena po rabacie} = \text{cena przed rabatem} - \text{kwota rabatu}$.
 - Wyniki są wyświetlane w podsumowaniu z dokładnością do dwóch miejsc po przecinku.
 - W przypadku nieprawidłowego rabatu (np. >50 lub wartość nieliczbowa) wyświetlany jest MessageBox z komunikatem i obliczenia nie są wykonywane.

4. Kryteria Oceniania

- R.3.1: Okno aplikacji ma tytuł zawierający numer PESEL zdającego.
- R.3.2: Zaimplementowano wybór rozmiaru pizzy za pomocą RadioButton oraz dodatków za pomocą CheckBox.
- R.3.3: Program waliduje poprawność pola rabatu (liczba z zakresu 0–50).
- R.3.4: Program poprawnie oblicza cenę przed rabatem na podstawie rozmiaru i dodatków.
- R.3.5: Program poprawnie oblicza kwotę rabatu i cenę po rabacie.
- R.3.6: W przypadku błędnych danych rabatu wyświetlany jest komunikat błędu i obliczenia nie są wykonywane.
- R.3.7: Wynik jest wyświetlany w czytelnej formie (podział na cenę przed rabatem, rabat i cenę po rabacie).

Zadanie 3

1. Tytuł Zadania

Aplikacja: Kalkulator Średniej Ocen

2. Opis Interfejsu (GUI)

- Tytuł okna: „Średnia ocen – PESEL zdającego”.
- Kontrolki:
 - Etykiety i TextBoxy dla pięciu przedmiotów, np.:
 - * „Język polski”
 - * „Matematyka”
 - * „Język obcy”
 - * „Informatyka”
 - * „WF”
 - Przycisk „Oblicz średnią”.
 - Etykieta/Label z wynikiem „Średnia:”.
 - Etykieta „Klasyfikacja:” (opis słowny, np. „dostateczna”, „dobra”, „bardzo dobra”).
- Każde pole przyjmuje oceny z zakresu 1–6 (dopuszczalne również wartości dziesiętne, np. 4,5).
- Początkowo pola wynikowe są puste.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Oblicz średnią”:
 - Program sprawdza:
 - * Czy wszystkie pola są wypełnione.
 - * Czy wpisane wartości są liczbami z zakresu 1–6.
 - Oblicza średnią arytmetyczną z pięciu ocen (dokładność do dwóch miejsc po przecinku).
 - Na podstawie średniej przypisuje klasyfikację:
 - * $< 2,0$ – „niedostateczna”
 - * $2,0$ – $3,49$ – „dostateczna”
 - * $3,5$ – $4,49$ – „dobra”
 - * $4,5$ – $5,49$ – „bardzo dobra”
 - * $\geq 5,5$ – „celująca”.
 - Wyniki są wyświetlane w odpowiednich etykietach.
 - Przy błędnych danych (puste pole, liczba spoza zakresu, tekst zamiast liczby) wyświetlany jest MessageBox.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Interfejs zawiera pięć pól do wprowadzania ocen i etykiety wynikowe.
- R.3.3: Program waliduje, że wszystkie wprowadzone wartości są liczbami z zakresu 1–6.
- R.3.4: Program poprawnie oblicza średnią ocen.
- R.3.5: Program poprawnie przypisuje klasyfikację opisową na podstawie uzyskanej średniej.
- R.3.6: Przy niepoprawnych danych wejściowych wyświetlany jest komunikat błędu, a wyniki nie są aktualizowane.

Zadanie 4

1. Tytuł Zadania

Aplikacja: Kalkulator BMI

2. Opis Interfejsu (GUI)

- Tytuł: „Kalkulator BMI – PESEL zdającego”.
- Kontrolki:
 - Label i TextBox „Wzrost (cm)”.
 - Label i TextBox „Masa ciała (kg)”.
 - Przycisk „Oblicz BMI”.
 - Label lub TextBox (tylko do odczytu) „Wynik BMI”.
 - Label „Kategoria:” (np. „Niedowaga”, „Prawidłowa”, „Nadwaga”, „Otyłość”).
- Przycisk aktywny od początku.
- Pola wynikowe początkowo puste.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Oblicz BMI”:
 - Program weryfikuje:
 - * Czy oba pola (wzrost i masa) są wypełnione.
 - * Czy wzrost i waga są dodatnimi liczbami.
 - * Czy wzrost jest w realistycznym zakresie (np. 100–250 cm).
 - Oblicza BMI według wzoru:
$$\text{BMI} = \text{masa}(\text{kg}) / (\text{wzrost}(\text{m}))^2$$
(przeliczenie cm na metry: $\text{wzrost_m} = \text{wzrost_cm} / 100$).
 - Wyświetla wartość BMI z dwoma miejscami po przecinku.
 - Przypisuje kategorię:
 - * $< 18,5$ – „Niedowaga”

- * 18,5–24,9 – „Prawidłowa”
 - * 25,0–29,9 – „Nadwaga”
 - * $\geq 30,0$ – „Otyłość”.
- Przy błędnych danych wejściowych wyświetlany jest MessageBox, a obliczenia nie są wykonywane.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Zaimplementowano wymagane pola i przycisk.
- R.3.3: Program sprawdza poprawność danych wejściowych (puste pola, dodatnie liczby, zakres wzrostu).
- R.3.4: Program poprawnie oblicza wartość BMI.
- R.3.5: Program poprawnie klasyfikuje wynik do jednej z kategorii.
- R.3.6: W przypadku błędnych danych pokazywany jest komunikat (MessageBox).

Zadanie 5

1. Tytuł Zadania

Aplikacja: Konwerter Walut

2. Opis Interfejsu (GUI)

- Tytuł: „Konwerter walut – PESEL zdającego”.
- Kontrolki:
 - TextBox „Kwota w PLN”.
 - ComboBox „Waluta docelowa” z pozycjami: „EUR”, „USD”, „GBP”.
 - Label z kursami (stałe, np. EUR = 4,50; USD = 4,10; GBP = 5,20).
 - Przycisk „Przelicz”.
 - Label lub TextBox tylko do odczytu „Wynik:”.
- Domyślnie zaznaczona waluta: „EUR”.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Przelicz”:
 - Program sprawdza, czy kwota jest podana i czy jest dodatnią liczbą.
 - Na podstawie wybranej waluty pobiera stały kurs (zakodowany w programie).
 - Oblicza:

$$\text{kwota_w_walucie} = \text{kwota_PLN} / \text{kurs}.$$
 - Wynik wyświetlany jest z dokładnością do dwóch miejsc po przecinku wraz z kodem waluty (np. „123,45 EUR”).
 - Przy błędnych danych wejściowych (puste pole, tekst zamiast liczby) wyświetlany jest MessageBox.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Aplikacja zawiera TextBox dla kwoty, ComboBox z trzema walutami, przycisk i pole wyniku.
- R.3.3: Program waliduje, czy wprowadzona kwota jest dodatnią liczbą.
- R.3.4: Program wykorzystuje odpowiedni kurs w zależności od wybranej waluty.
- R.3.5: Program poprawnie oblicza i wyświetla wynik konwersji.
- R.3.6: Program reaguje komunikatem błędu na niepoprawne dane wejściowe.

Zadanie 6

1. Tytuł Zadania

Aplikacja: Lista Zadań (ToDo) z Priorytetem

2. Opis Interfejsu (GUI)

- Tytuł: „Lista zadań – PESEL zdającego”.
- Kontrolki:
 - TextBox „Nazwa zadania”.
 - ComboBox „Priorytet” z wartościami: „Niski”, „Średni”, „Wysoki”.
 - Przycisk „Dodaj zadanie”.
 - ListBox wyświetlający dodane zadania w formacie np. „[Wysoki] Kupić bilety”.
 - Przycisk „Usuń wybrane”.
 - Przycisk „Wyczyść listę”.
- Początkowo ComboBox ustawiony na „Średni”.
- Przycisk „Usuń wybrane” jest nieaktywny, gdy w ListBox nie ma zaznaczonego elementu.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Dodaj zadanie”:
 - Program sprawdza, czy TextBox z nazwą zadania nie jest pusty.
 - Tworzy tekst zadania w formacie: „[Priorytet] NazwaZadania”.
 - Dodaje element do ListBox.
 - Czyści TextBox z nazwą zadania.
- Po zaznaczeniu elementu w ListBox przycisk „Usuń wybrane” staje się aktywny.
- Po kliknięciu „Usuń wybrane”:
 - Zaznaczone zadanie zostaje usunięte z ListBox.

- Po kliknięciu „Wyczyść listę”:
 - Wszystkie zadania są usuwane z ListBox.
 - Przycisk „Usuń wybrane” zostaje dezaktywowany.
- Jeżeli użytkownik próbuje dodać puste zadanie, wyświetlany jest MessageBox z informacją o błędzie.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Interfejs zawiera pola do wprowadzania zadania, wybór priorytetu, ListBox oraz wymagane przyciski.
- R.3.3: Program nie pozwala dodać zadania o pustej nazwie (komunikat błędu).
- R.3.4: Program dodaje zadania do listy w prawidłowym formacie z priorytetem.
- R.3.5: Program umożliwia usunięcie zaznaczonego zadania i wyczyszczenie całej listy.
- R.3.6: Stan przycisku „Usuń wybrane” zmienia się poprawnie w zależności od zaznaczenia w liście.

Zadanie 7

1. Tytuł Zadania

Aplikacja: Weryfikator Hasła

2. Opis Interfejsu (GUI)

- Tytuł: „Weryfikator hasła – PESEL zdającego”.
- Kontrolki:
 - TextBox „Hasło” (w trybie ukrywania znaków – właściwość PasswordChar lub odpowiedni atrybut).
 - Przycisk „Sprawdź hasło”.
 - Trzy CheckBox (tylko do odczytu lub zablokowane przed zmianą przez użytkownika):
 - * „Minimum 8 znaków”
 - * „Zawiera cyfrę”
 - * „Zawiera wielką literę”
 - Label „Ocena hasła:” (np. „słabe”, „średnie”, „silne”).
- Początkowo wszystkie CheckBox są odznaczone.
- Pole „Ocena hasła” jest puste.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Sprawdź hasło”:
 - Program sprawdza warunki:
 - * Długość hasła ≥ 8 znaków.
 - * Hasło zawiera co najmniej jedną cyfrę (0–9).
 - * Hasło zawiera co najmniej jedną wielką literę (A–Z).
 - Dla każdego spełnionego warunku zaznacza odpowiedni CheckBox.
 - Na podstawie liczby spełnionych warunków określa ocenę:
 - * 0–1 warunek – „słabe”
 - * 2 warunki – „średnie”
 - * 3 warunki – „silne”.
 - Ocena jest wyświetlana w etykiecie „Ocena hasła”.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Pole hasła działa w trybie ukrywania wprowadzanych znaków.
- R.3.3: Program poprawnie weryfikuje długość hasła.
- R.3.4: Program poprawnie wykrywa obecność przynajmniej jednej cyfry i wielkiej litery.
- R.3.5: CheckBoxy odzwierciedlają aktualny stan spełnienia poszczególnych warunków.
- R.3.6: Program poprawnie wyznacza ocenę hasła na podstawie liczby spełnionych kryteriów.

Zadanie 8

1. Tytuł Zadania

Aplikacja: Kalkulator Opłaty Parkingowej

2. Opis Interfejsu (GUI)

- Tytuł: „Parking – PESEL zdającego”.
- Kontrolki:
 - TextBox „Czas postoju (w godzinach, np. 1,5)”.
 - TextBox „Stawka za godzinę (zł)”.
 - CheckBox „Karta stałego klienta (10% rabatu)”.
 - Przycisk „Oblicz opłatę”.
 - Label lub TextBox tylko do odczytu „Kwota do zapłaty”.
- Początkowo CheckBox jest odznaczony.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Oblicz opłatę”:
 - Program sprawdza, czy pola są wypełnione oraz czy wprowadzono dodatnie liczby.
 - Oblicza opłatę:
 - * $\text{kwota} = \text{czas} \times \text{stawka}$.
 - Jeżeli zaznaczono „Karta stałego klienta”, stosuje rabat 10%:
 - * $\text{kwota} = \text{kwota} \times 0,9$.
 - Wynik wyświetla z dwoma miejscami po przecinku.
 - W przypadku błędnych danych wejściowych wyświetlany jest MessageBox.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Interfejs zawiera pola do wprowadzenia czasu i stawki, CheckBox rabatu, przycisk i pole wyniku.
- R.3.3: Program waliduje, że czas i stawka są dodatnimi liczbami.
- R.3.4: Program poprawnie oblicza opłatę bez rabatu.
- R.3.5: Program poprawnie uwzględnia rabat 10% dla stałego klienta.
- R.3.6: Przy błędnych danych wejściowych wyświetlany jest komunikat o błędzie.

Zadanie 9

1. Tytuł Zadania

Aplikacja: Konwerter Temperatur

2. Opis Interfejsu (GUI)

- Tytuł: „Konwerter temperatur – PESEL zdającego”.
- Kontrolki:
 - TextBox „Wartość temperatury”.
 - ComboBox „Jednostka wejściowa” z opcjami: „Celsjusz”, „Fahrenheit”.
 - ComboBox „Jednostka wyjściowa” z tymi samymi opcjami.
 - Przycisk „Przelicz”.
 - Label lub TextBox tylko do odczytu „Wynik”.
- Domyślnie: jednostka wejściowa = „Celsjusz”, jednostka wyjściowa = „Fahrenheit”.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Przelicz”:
 - Program sprawdza, czy podano liczbę (może być ujemna).
 - W zależności od kombinacji jednostek:
 - * Celsjusz \rightarrow Fahrenheit:
$$F = C \times \frac{9}{5} + 32.$$
 - * Fahrenheit \rightarrow Celsjusz:
$$C = (F - 32) \times \frac{5}{9}.$$
 - * Celsjusz \rightarrow Celsjusz lub Fahrenheit \rightarrow Fahrenheit:
wynik równy wartości wejściowej.
 - Wynik wyświetlany z dwoma miejscami po przecinku oraz jednostką.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Interfejs umożliwia wybór jednostek wejściowej i wyjściowej oraz wprowadzenie wartości.
- R.3.3: Program sprawdza poprawność danych (czy wprowadzono liczbę).
- R.3.4: Program poprawnie przelicza temperaturę między skalami Celsjusza i Fahrenheita.
- R.3.5: Dla tej samej skali na wejściu i wyjściu wynik jest równy wartości wejściowej.
- R.3.6: Wynik jest wyświetlany z odpowiednią jednostką.

Zadanie 10

1. Tytuł Zadania

Aplikacja: Analizator Tekstu

2. Opis Interfejsu (GUI)

- Tytuł: „Analizator tekstu – PESEL zdającego”.
- Kontrolki:
 - Wielowierszowy TextBox do wprowadzania tekstu.
 - Przycisk „Analizuj”.
 - Trzy Label lub TextBox tylko do odczytu:
 - * „Liczba znaków (łącznie):”
 - * „Liczba znaków bez spacji:”
 - * „Liczba słów:”.
- Początkowo pola wynikowe są puste.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Analizuj”:
 - Program pobiera tekst z TextBox.
 - Oblicza:
 - * Liczbę wszystkich znaków (długość łańcucha).
 - * Liczbę znaków bez spacji – pomijając znak spacji (' '); inne białe znaki mogą być liczone.
 - * Liczbę słów – przyjmując, że słowa są oddzielone co najmniej jedną spacją; puste fragmenty (powstałe np. przy wielu spacjach obok siebie) są pomijane.
 - Wyniki wyświetla w odpowiednich polach.
 - Jeżeli TextBox jest pusty, wyświetlany jest MessageBox z informacją, że brak tekstu do analizy.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Aplikacja udostępnia wielowierszowy TextBox i trzy pola wynikowe.
- R.3.3: Program prawidłowo liczy liczbę wszystkich znaków.
- R.3.4: Program prawidłowo liczy liczbę znaków bez spacji.
- R.3.5: Program prawidłowo wyznacza liczbę słów z uwzględnieniem wielu spacji.
- R.3.6: Program reaguje komunikatem na próbę analizy pustego tekstu.

Zadanie 11

1. Tytuł Zadania

Aplikacja: Prosty Szyfrator (Szyfr Cezara)

2. Opis Interfejsu (GUI)

- Tytuł: „Szyfr Cezara – PESEL zdającego”.
- Kontrolki:
 - TextBox „Tekst jawny”.
 - TextBox „Przesunięcie” (liczba całkowita z zakresu 1–10).
 - Przycisk „Zaszyfruj”.
 - Wielowierszowy TextBox „Tekst zaszyfrowany” (tylko do odczytu).
- Początkowo pole „Tekst zaszyfrowany” jest puste.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Zaszyfruj”:
 - Program sprawdza:
 - * Czy tekst jawny nie jest pusty.
 - * Czy przesunięcie jest liczbą z zakresu 1–10.
 - Szyfrowanie:
 - * Szyfrowane są tylko litery alfabetu łacińskiego (A–Z, a–z).
 - * Inne znaki (spacje, cyfry, znaki interpunkcyjne) pozostają bez zmian.
 - * Dla liter:
 - Znak jest przesuwany w alfabecie o podaną liczbę pozycji z zawijaniem (np. 'Z' przy przesunięciu 1 → 'A').
 - Zachowana jest wielkość liter (mała/duża).
 - Wynik wstawiany jest do pola „Tekst zaszyfrowany”.
 - Przy błędnych danych przesunięcia lub pustym tekście wyświetlany jest MessageBox.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Interfejs zawiera pola dla tekstu jawnego, przesunięcia i tekstu zaszyfrowanego.
- R.3.3: Program waliduje przesunięcie (liczba 1–10) i niepusty tekst.
- R.3.4: Program poprawnie szyfruje litery z zachowaniem wielkości i zawijaniem alfabetu.
- R.3.5: Program pozostawia nieliterowe znaki bez zmian.
- R.3.6: Tekst zaszyfrowany jest wyświetlany w odpowiednim polu.

Zadanie 12

1. Tytuł Zadania

Aplikacja: Kalkulator Raty Kredytu (uproszczony)

2. Opis Interfejsu (GUI)

- Tytuł: „Rata kredytu – PESEL zdającego”.
- Kontrolki:
 - TextBox „Kwota kredytu (zł)”.
 - TextBox „Liczba rat (miesiące)”.
 - TextBox „Oprocentowanie roczne (%)”.
 - Przycisk „Oblicz ratę”.
 - Label lub TextBox tylko do odczytu „Rata miesięczna:”.
- Początkowo pole wynikowe jest puste.

3. Opis Działania Aplikacji i Logiki

Przyjmujemy uproszczony model:

$\text{rata równa} = (\text{kwota} + \text{odsetki}) / \text{liczba rat},$

$\text{odsetki} = \text{kwota} \times (\text{oprocentowanie_roczne} / 100).$

- Po kliknięciu „Oblicz ratę”:
 - Program sprawdza:
 - * Czy wszystkie pola są wypełnione.
 - * Czy kwota i liczba rat są dodatnimi liczbami (liczba rat całkowita ≥ 1).
 - * Czy oprocentowanie jest liczbą ≥ 0 .
 - Oblicza:
 - * $\text{Odsetki} = \text{kwota} \times (\text{oprocentowanie} / 100).$
 - * $\text{Całkowita kwota do spłaty} = \text{kwota} + \text{odsetki}.$
 - * $\text{Rata miesięczna} = \text{całkowita kwota} / \text{liczba rat}.$
 - Wynik wyświetlany z dwoma miejscami po przecinku.
 - Przy błędnych danych wyświetlany jest MessageBox.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Interfejs zawiera wymagane pola wejściowe i pole wyniku.
- R.3.3: Program poprawnie waliduje dane ($\text{kwota} > 0$, $\text{liczba rat} \geq 1$, $\text{oprocentowanie} \geq 0$).
- R.3.4: Program poprawnie oblicza odsetki i całkowitą kwotę do spłaty.
- R.3.5: Program poprawnie oblicza ratę miesięczną.
- R.3.6: Program zgłasza błędy danych wejściowych poprzez MessageBox.

Zadanie 13

1. Tytuł Zadania

Aplikacja: Planer Budżetu Domowego

2. Opis Interfejsu (GUI)

- Tytuł: „Budżet domowy – PESEL zdającego”.
- Kontrolki:
 - TextBox „Budżet miesięczny (zł)”.
 - Trzy TextBoxy z etykietami:
 - * „Czynsz”
 - * „Jedzenie”
 - * „Transport”
 - Przycisk „Podsumuj”.

- Dwa Label lub TextBox tylko do odczytu:
 - * „Suma wydatków:”
 - * „Różnica (budżet – wydatki):”
- Label „Status:” (np. „w budżecie”, „przekroczono budżet”).
- Początkowo pola wynikowe i status są puste.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Podsumuj”:
 - Program sprawdza, czy wszystkie pola zawierają liczby ≥ 0 .
 - Oblicza:
 - * Suma wydatków = czynsz + jedzenie + transport.
 - * Różnica = budżet – suma wydatków.
 - Wyświetla sumę i różnicę z dwoma miejscami po przecinku.
 - Jeżeli różnica $\geq 0 \rightarrow$ „Status: w budżecie”.
 - Jeżeli różnica $< 0 \rightarrow$ „Status: przekroczono budżet”.
 - Przy błędnych danych (tekst, puste pola) wyświetlany jest MessageBox.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Interfejs zawiera pola dla budżetu i trzech kategorii wydatków oraz pola wynikowe.
- R.3.3: Program waliduje, że wszystkie wartości są liczbami (≥ 0).
- R.3.4: Program poprawnie oblicza sumę wydatków.
- R.3.5: Program poprawnie oblicza różnicę między budżetem a wydatkami.
- R.3.6: Program wyświetla poprawny status („w budżecie” lub „przekroczono budżet”).

Zadanie 14

1. Tytuł Zadania

Aplikacja: Generator Haseł

2. Opis Interfejsu (GUI)

- Tytuł: „Generator haseł – PESEL zdającego”.
- Kontrolki:
 - NumericUpDown lub TextBox „Długość hasła” (liczba całkowita 4–20).
 - CheckBox „Małe litery”.
 - CheckBox „Duże litery”.
 - CheckBox „Cyfry”.

- CheckBox „Znaki specjalne (!@#\$. . .)”.
- Przycisk „Generuj”.
- TextBox „Wygenerowane hasło” (tylko do odczytu).
- Domyślnie zaznaczone: „Małe litery” i „Cyfry”.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Generuj”:
 - Program sprawdza:
 - * Czy wybrano co najmniej jeden rodzaj znaków (co najmniej jeden CheckBox jest zaznaczony).
 - * Czy długość hasła jest w przedziale 4–20.
 - Buduje zbiór dozwolonych znaków na podstawie zaznaczonych opcji:
 - * Małe litery: a–z.
 - * Duże litery: A–Z.
 - * Cyfry: 0–9.
 - * Znaki specjalne: np. !@#% ^ & *.
 - Losuje znaki i tworzy hasło o zadanej długości.
 - Wyświetla hasło w TextBox „Wygenerowane hasło”.
 - Przy błędnych danych (np. brak zaznaczonych typów znaków, niepoprawna długość) wyświetlany jest MessageBox.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Interfejs zawiera wybór długości oraz typów znaków oraz pole z wynikiem.
- R.3.3: Program waliduje, że długość hasła mieści się w podanym zakresie.
- R.3.4: Program sprawdza, czy wybrano co najmniej jeden typ znaków.
- R.3.5: Program generuje hasło z losowych znaków z odpowiedniego zbioru.
- R.3.6: Wygenerowane hasło ma dokładnie zadaną długość.

Zadanie 15

1. Tytuł Zadania

Aplikacja: Rezerwacja Miejsc w Kinie (prosty model)

2. Opis Interfejsu (GUI)

- Tytuł: „Rezerwacja kina – PESEL zdającego”.
- Kontrolki:
 - ListBox z listą miejsc, np. „A1”, „A2”, ..., „A5”.

- Przycisk „Zarezerwuj miejsce”.
- Przycisk „Anuluj rezerwację”.
- Label „Status:” (informacja o ostatniej operacji).
- Początkowo:
 - Wszystkie miejsca są dostępne (niezarezerwowane).
 - Przycisk „Anuluj rezerwację” jest nieaktywny.

3. Opis Działania Aplikacji i Logiki

- Użytkownik wybiera miejsce z ListBox.
- Po kliknięciu „Zarezerwuj miejsce”:
 - Jeżeli dane miejsce nie jest jeszcze zarezerwowane:
 - * Zostaje oznaczone jako zarezerwowane (np. zmiana tekstu na „A1 (zajęte)”).
 - * W Label „Status” wyświetla się np. „Zarezerwowano miejsce A1”.
 - * Przycisk „Anuluj rezerwację” staje się aktywny.
 - Jeżeli miejsce jest już zarezerwowane:
 - * Wyświetlany jest MessageBox informujący, że miejsce jest już zajęte.
- Po kliknięciu „Anuluj rezerwację”:
 - Jeżeli aktualnie zaznaczone miejsce jest zarezerwowane:
 - * Status zmienia się na dostępne (usunięcie dopisku „(zajęte)”).
 - * Label „Status” wyświetla np. „Anulowano rezerwację A1”.
 - Jeżeli zaznaczone miejsce nie jest zarezerwowane:
 - * Wyświetlany jest MessageBox o braku rezerwacji.
- Jeżeli użytkownik nie wybierze żadnego miejsca, a kliknie któryś z przycisków, program może wyświetlić MessageBox z informacją o konieczności wyboru miejsca.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Interfejs zawiera ListBox z miejscami oraz przyciski „Zarezerwuj” i „Anuluj”.
- R.3.3: Program poprawnie oznacza miejsce jako zarezerwowane.
- R.3.4: Program nie pozwala ponownie zarezerwować już zajętego miejsca (komunikat błędu).
- R.3.5: Program umożliwia anulowanie istniejącej rezerwacji i przywrócenie statusu miejsca.
- R.3.6: Label „Status” jest aktualizowany po każdej udanej operacji.

Zadanie 16

1. Tytuł Zadania

Aplikacja: Kalkulator Kosztu Paliwa

2. Opis Interfejsu (GUI)

- Tytuł: „Koszt paliwa – PESEL zdającego”.
- Kontrolki:
 - TextBox „Długość trasy (km)”.
 - TextBox „Średnie spalanie (l/100 km)”.
 - TextBox „Cena paliwa (zł/l)”.
 - Przycisk „Oblicz”.
 - Dwa Label lub TextBox tylko do odczytu:
 - * „Zużyte paliwo (l):”
 - * „Koszt paliwa (zł):”.
- Początkowo pola wynikowe są puste.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Oblicz”:
 - Program sprawdza, czy wszystkie pola są wypełnione i czy podano dodatnie liczby.
 - Oblicza:
 - * $\text{Zużyte paliwo} = (\text{długość_trasy} \times \text{spalanie}) / 100.$
 - * $\text{Koszt paliwa} = \text{zużyte_paliwo} \times \text{cena_paliwa}.$
 - Wyniki wyświetla z dwoma miejscami po przecinku.
 - W przypadku błędnych danych wejściowych wyświetlany jest MessageBox.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Interfejs zawiera pola wejściowe dla trasy, spalania i ceny oraz pola wynikowe.
- R.3.3: Program waliduje, że wszystkie wartości są dodatnimi liczbami.
- R.3.4: Program poprawnie oblicza zużycie paliwa.
- R.3.5: Program poprawnie oblicza koszt paliwa.
- R.3.6: Wyniki są prezentowane w czytelnej formie z dwoma miejscami po przecinku.

Zadanie 17

1. Tytuł Zadania

Aplikacja: Formularz Rejestracji Uczestnika Szkolenia

2. Opis Interfejsu (GUI)

- Tytuł: „Rejestracja szkolenia – PESEL zdającego”.
- Kontrolki:
 - TextBox „Imię”.
 - TextBox „Nazwisko”.
 - TextBox „E-mail”.
 - ComboBox „Rodzaj szkolenia” (np. „Online”, „Stacjonarne”).
 - CheckBox „Akceptuję regulamin”.
 - Przycisk „Zarejestruj”.
 - Wielowierszowy TextBox „Podsumowanie” (tylko do odczytu).
- Początkowo CheckBox jest odznaczony, a przycisk „Zarejestruj” nieaktywny.

3. Opis Działania Aplikacji i Logiki

- Przycisk „Zarejestruj”:
 - Jest aktywowany dopiero po zaznaczeniu CheckBox „Akceptuję regulamin”.
- Po kliknięciu „Zarejestruj”:
 - Program sprawdza:
 - * Czy pola „Imię”, „Nazwisko” i „E-mail” nie są puste.
 - * Czy e-mail zawiera znak „@” i kropkę (prosta walidacja).
 - Jeżeli dane są poprawne:
 - * W TextBox „Podsumowanie” wyświetla się informacja, np.: „Zarejestrowano: Imię Nazwisko, e-mail: ..., szkolenie: ...”.
 - W przypadku błędnych danych wyświetlany jest MessageBox z opisem błędu.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Przycisk „Zarejestruj” jest nieaktywny przed zaznaczeniem akceptacji regulaminu.
- R.3.3: Program waliduje niepustość pól „Imię”, „Nazwisko”, „E-mail”.
- R.3.4: Program sprawdza podstawową poprawność adresu e-mail (obecność „@” i kropki).
- R.3.5: Po poprawnej rejestracji w polu podsumowania wyświetlane są wszystkie wymagane informacje.
- R.3.6: Przy błędnych danych lub braku akceptacji regulaminu pojawia się odpowiedni komunikat.

Zadanie 18

1. Tytuł Zadania

Aplikacja: Kalendarz Urodzin – Wiek i Dni do Urodzin

2. Opis Interfejsu (GUI)

- Tytuł: „Kalendarz urodzin – PESEL zdającego”.
- Kontrolki:
 - Kontrolka wyboru daty (DateTimePicker lub analogiczna) „Data urodzenia”.
 - Przycisk „Oblicz”.
 - Dwa Label lub TextBox tylko do odczytu:
 - * „Wiek (w latach):”
 - * „Dni do najbliższych urodzin:”.
- Początkowo pola wynikowe są puste.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Oblicz”:
 - Program pobiera aktualną datę systemową.
 - Oblicza wiek w pełnych latach (różnica lat, uwzględniając, czy urodziny już były w danym roku).
 - Oblicza datę najbliższych urodzin:
 - * Jeżeli urodziny w bieżącym roku jeszcze nie minęły, używa bieżącego roku.
 - * W przeciwnym razie używa roku następnego.
 - Oblicza liczbę dni między dzisiejszą datą a najbliższymi urodzinami.
 - Wyniki wyświetla w polach wynikowych.
 - Można założyć, że użytkownik nie wybierze daty z przyszłości; opcjonalnie w takim przypadku program wyświetla MessageBox z błędem.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Interfejs zawiera kontrolkę wyboru daty oraz dwa pola wynikowe.
- R.3.3: Program poprawnie oblicza wiek w pełnych latach.
- R.3.4: Program poprawnie wyznacza datę najbliższych urodzin (bieżący lub następny rok).
- R.3.5: Program poprawnie oblicza liczbę dni do najbliższych urodzin.
- R.3.6: Wybranie daty przyszłej (jeśli obsługiwane) powoduje wyświetlenie komunikatu o błędzie.

Zadanie 19

1. Tytuł Zadania

Aplikacja: Mini-Quiz Jednokrotnego Wyboru

2. Opis Interfejsu (GUI)

- Tytuł: „Mini-quiz – PESEL zdającego”.
- Kontrolki:
 - Trzy pytania (Label), każde z przypisaną grupą RadioButton (po 3 odpowiedzi do każdego pytania).
 - Przycisk „Sprawdź wynik”.
 - Label lub TextBox tylko do odczytu „Wynik: X/3”.
- Początkowo żaden RadioButton nie jest zaznaczony, pole wyniku jest puste.

3. Opis Działania Aplikacji i Logiki

- Użytkownik zaznacza po jednej odpowiedzi w każdej grupie.
- Po kliknięciu „Sprawdź wynik”:
 - Program sprawdza, czy w każdej grupie wybrano jedną odpowiedź:
 - * Jeżeli nie, wyświetlany jest MessageBox z informacją o konieczności odpowiedzi na wszystkie pytania.
 - Jeżeli wszystkie odpowiedzi są zaznaczone:
 - * Program porównuje zaznaczone odpowiedzi z kluczem poprawnych odpowiedzi (zakodowanym w programie).
 - * Liczy liczbę poprawnych odpowiedzi (0–3).
 - * Wyświetla wynik w formacie „Wynik: X/3”.
 - * Opcjonalnie może wyświetlić krótki komentarz (np. „słabo”, „dobrze”, „bardzo dobrze”).

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Dla każdego pytania zrealizowano grupę RadioButton z trzema odpowiedziami.
- R.3.3: Program wymusza udzielenie odpowiedzi na wszystkie pytania (w przypadku braków wyświetla komunikat).
- R.3.4: Program prawidłowo porównuje odpowiedzi z kluczem.
- R.3.5: Program poprawnie zlicza liczbę poprawnych odpowiedzi.
- R.3.6: Wynik prezentowany jest w formacie „X/3”.

Zadanie 20

1. Tytuł Zadania

Aplikacja: Generator Liczb Lotto

2. Opis Interfejsu (GUI)

- Tytuł: „Generator Lotto – PESEL zdającego”.
- Kontrolki:
 - Przycisk „Wylosuj liczby”.
 - Label lub TextBox tylko do odczytu z wynikiem (np. 6 liczb).
 - Opcjonalnie: CheckBox „Posortuj rosnąco”.
- Po uruchomieniu pola wynikowe są puste, CheckBox jest odznaczony.

3. Opis Działania Aplikacji i Logiki

- Po kliknięciu „Wylosuj liczby”:
 - Program losuje 6 różnych liczb całkowitych z zakresu 1–49.
 - Jeżeli zaznaczono „Posortuj rosnąco”, sortuje wylosowane liczby rosnąco.
 - Wyświetla liczby w jednym polu (np. rozdzielone przecinkami: „5, 12, 18, 33, 42, 49”).
 - Każde kliknięcie generuje nowy zestaw liczb.

4. Kryteria Oceniania

- R.3.1: Tytuł okna zawiera numer PESEL zdającego.
- R.3.2: Program losuje dokładnie 6 liczb z zakresu 1–49.
- R.3.3: Program zapewnia, że wylosowane liczby są unikalne (brak powtórzeń).
- R.3.4: Po zaznaczeniu opcji sortowania liczby są wyświetlane w kolejności rosnącej.
- R.3.5: Wynik jest prezentowany w czytelnej formie (liczby oddzielone np. przecinkami).
- R.3.6: Każde kliknięcie przycisku generuje nowy, niezależny zestaw liczb.