

Nazwa kwalifikacji: **Projektowanie, programowanie i testowanie aplikacji**

Symbol kwalifikacji: **INF.04**

Numer zadania: **01**

Wersja arkusza: **SG**

Wypełnia zdający

Numer PESEL zdającego*

--	--	--	--	--	--	--	--	--	--	--	--	--

Numer stanowiska

--	--	--

Miejsce na naklejkę z numerem
PESEL i z kodem ośrodka

Czas trwania egzaminu: **180** minut

INF.04-01-26.06-SG

EGZAMIN ZAWODOWY

Rok 2026

CZĘŚĆ PRAKTYCZNA

**PODSTAWA PROGRAMOWA
2019**

Instrukcja dla zdającego

1. Na pierwszej stronie arkusza egzaminacyjnego wpisz w oznaczonym miejscu swój numer PESEL*, numer stanowiska i naklej naklejkę** z numerem PESEL i z kodem ośrodka.
2. Sprawdź, czy arkusz egzaminacyjny zawiera 6 stron i nie zawiera błędów. Ewentualny brak stron lub inne usterki zgłoś przez podniesienie ręki przewodniczącemu zespołu nadzorującego.
3. Zapoznaj się z treścią zadania oraz stanowiskiem egzaminacyjnym. Masz na to 10 minut. Czas ten nie jest wliczany do czasu trwania egzaminu.
4. Czas rozpoczęcia i zakończenia pracy zapisze w widocznym miejscu przewodniczący zespołu nadzorującego.
5. Wykonaj samodzielnie zadanie egzaminacyjne. Przestrzegaj zasad bezpieczeństwa i organizacji pracy.
6. Po zakończeniu wykonania zadania pozostaw arkusz egzaminacyjny z rezultatami wykonania zadania na swoim stanowisku lub w miejscu wskazanym przez przewodniczącego zespołu nadzorującego.
7. Po uzyskaniu zgody zespołu nadzorującego możesz opuścić salę/miejsce przeprowadzania egzaminu.

Powodzenia!

* w przypadku braku numeru PESEL – seria i numer paszportu lub innego dokumentu potwierdzającego tożsamość

** w przypadku otrzymania naklejki

Zadanie egzaminacyjne

UWAGA: numerem zdającego jest w zadaniu nazywany numer, którym został podpisany arkusz egzaminacyjny (PESEL lub, w przypadku jego braku, numer paszportu).

Wykonaj aplikacje konsolową i webową oraz ich dokumentację. Aplikacje wykonaj według wskazań podanych w części I i II zadania, a dokumentację zgodnie z opisem w części III. Wykorzystaj konto **Egzamin** bez hasła.

Utwórz folder i nazwij go numerem zdającego. W folderze utwórz trzy podfoldery o nazwach: *konsolowa*, *webowa*, *dokumentacja*. Po wykonaniu zadania umieścisz w nich rezultaty swojej pracy.

Do wykonania aplikacji należy wykorzystać materiały z archiwum *Web1.7z* znajdującego się na pulpicie konta **Egzamin** i zabezpieczonego hasłem: ***LoGiny&H@sla8**

Część I. Aplikacja konsolowa

Za pomocą narzędzi do tworzenia aplikacji konsolowych zaimplementuj program do tworzenia bezpiecznych haseł.

Założenia aplikacji:

- Zastosowany obiektowy język programowania zgodny z zainstalowanym na stanowisku egzaminacyjnym: C++ lub C#, lub Java, lub Python
- Dowolne podejście: strukturalne lub obiektowe
- Zastosowane znaczące nazewnictwo, polskie albo angielskie
- Kod zapisany czytelnie, z zachowaniem zasad czystego formatowania
- Komunikacja z użytkownikiem jest prowadzona w sposób zrozumiały
- Nie można zastosować metody wbudowanej lub z funkcji bibliotecznych do generowania gotowego hasła

- Hasło przechowywane w zmiennej / polu typu napisowego
- Funkcja / metoda zwracająca wygenerowane 12-znakowe hasło według zasad:
 - Losowane są:
 - 3 małe litery z alfabetu łacińskiego (26 znaków: a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z)
 - 3 wielkie litery z alfabetu łacińskiego (26 znaków: A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z)
 - 2 litery ze znakami diakrytycznymi ze zbioru: ą, ę, ł, ń, ó, ś, ż, ź, Ą, Ę, Ł, Ń, Ó, Ś, Ż, Ź (patrz plik *znaki.txt*)
 - 2 cyfry z zakresu 0 – 9
 - 2 znaki specjalne z zestawu: !@#\$%^&*()_+
 - Każdy z wylosowanych 12 znaków jest umieszczany w losowym miejscu w hasle
 - Hasło jest zapisane w zmiennej napisowej i zwracane z funkcji
- Aplikacja w programie głównym generuje 5 haseł i wyświetla je ze znaczącym komunikatem
- Do kodu należy dołączyć dokumentację, która została opisana w części III zadania egzaminacyjnego

Kod aplikacji przygotuj do nagrania na płytę. W folderze *konsolowa* należy umieścić archiwum całego projektu o nazwie *konsola.zip*, i skopiowany z projektu plik z kodem źródłowym programu (modyfikowane pliki) oraz plik wykonywalny, jeżeli istnieje.

Część II. Aplikacja webowa

Z zastosowaniem dostępnego na stanowisku egzaminacyjnym frameworka Angular lub biblioteki React.js wykonaj aplikację internetową typu front-end obsługującą logowanie. Aplikacja jest uproszczoną wersją menedżera haseł, dlatego hasło jest zapisywane w sposób jawny. Działanie aplikacji przedstawiono na ilustracjach 1, 2, 3. W zależności od zastosowanego narzędzia wygląd aplikacji może nieznacznie się różnić.

Menedżer haseł - wykonał 000000000000

Adres WWW:
Adres strony WWW

Login:
Twój login

☒ Wygeneruj hasło
Hasło:

Zapisz

Adres:

Login:

Hasło automatycznie generowane

Ilustracja 1. Stan początkowy aplikacji Angular

Menedżer haseł - wykonał 000000000000

Adres WWW:
poczta.pl

Login:
Tomek

☒ Wygeneruj hasło
Hasło:

Zapisz

Adres: poczta.pl

Login: Tomek

Hasło automatycznie generowane

Ilustracja 2. Zapisanie danych, gdy pole wyboru jest zaznaczone

Menedżer haseł - wykonał 000000000000

Adres WWW:

Login:

☐ Wygeneruj hasło

Hasło:

Adres: poczta.pl

Login: Ola

Hasło: sYt@kU*&

Ilustracja 3. Zapisanie danych, gdy pole wyboru nie jest zaznaczone

Założenia aplikacji:

- Aplikacja składa się z jednego komponentu, w którym znajdują się formularz i podsumowanie danych z formularza
- Do utworzenia formularza osztylowanego biblioteką Bootstrap należy posłużyć się przykładem z pliku *szablon.html* wypakowanego z archiwum. Fragmenty kodu HTML należy zmodyfikować zgodnie ze składnią Angular lub React.js
- Komponent składa się z dwóch bloków
- Elementy bloku pierwszego:
 - Nagłówek pierwszego stopnia o treści „Menedżer haseł - wykonał”, dalej wstawiony numer zdającego
 - Formularz z trzema polami edycyjnymi, polem wyboru i przyciskiem „Zapisz”, wykonany zgodnie z ilustracją 1, stosujący bibliotekę Bootstrap
 - Pola edycyjne są powiązane z etykietami: „Adres WWW:”, „Login:”, „Hasło:”
 - Pola do wprowadzenia adresu WWW i loginu zawierają tekst podpowiedzi: "Adres strony WWW", "Twój login"
 - Pole wyboru jest domyślnie zaznaczone i pole do wpisania hasła jest nieaktywne
 - Pole do wpisania hasła umożliwia ukrywanie wpisywanych znaków
 - Gdy pole wyboru jest zaznaczone to pole do wpisania hasła jest nieaktywne (ilustracja 2), w przeciwnym wypadku jest aktywne (ilustracja 3)
 - Przycisk „Zapisz” jest koloru niebieskiego za sprawą odpowiedniej klasy Bootstrap
- Elementy bloku drugiego:
 - Elementy są wypełniane po zatwierdzeniu formularza przyciskiem
 - Trzy paragrafy ze stylem nadanym odpowiednimi klasami Bootstrap: margines wewnętrzny 0,5 rem. Dodatkowo tekst pierwszego paragrafu koloru niebieskiego, tekst jest wiodący (tabela 1)
 - Paragraf pierwszy zawiera napis: „Adres: <adres>”
 - Paragraf drugi zawiera napis: „Login: <login>”
 - Paragraf trzeci, gdy pole wyboru jest zaznaczone, zawiera napis „Hasło automatycznie generowane” (ilustracja 2), w przeciwnym wypadku napis „Hasło: <hasło>” (ilustracja 3)
 - W opisanych paragrafach nawiasy < >, np. <adres>, oznaczają dane pobrane z formularza

Kod aplikacji powinien być zapisany czytelnie, z zachowaniem zasad czystego formatowania, należy stosować znaczące nazewnictwo, polskie albo angielskie.

Podejmij próbę uruchomienia aplikacji w przeglądarce. Informacje dotyczące zrzutów ekranu znajdują się w części III zadania egzaminacyjnego.

Kod aplikacji przygotuj do nagrania na płytę. W folderze *webowa* powinno znaleźć się archiwum całego folderu projektu o nazwie *web.zip* oraz skopiowane pliki z kodem źródłowym, które były modyfikowane.

Część III. Dokumentacja aplikacji

Wykonaj dokumentację do aplikacji utworzonych na egzaminie.

W kodzie źródłowym aplikacji konsolowej za pomocą komentarza utwórz jeden nagłówek do dowolnej funkcji / metody według wzoru z listingu 1. Komentarz powinien znaleźć się nad lub pod nazwą funkcji. W miejscu nawiasów <> należy podać odpowiednie opisy. W miejscu autor należy podać numer zdającego.

UWAGA: Dokumentację funkcji należy umieścić w komentarzu (wieloliniowym lub kilku jednoliniowych). Znajdujący się w listingu 1 wzór dokumentacji jest bez znaków początku i końca komentarza, gdyż te są różne dla różnych języków programowania. W przypadku podejścia obiektowego zamiast słowa „funkcji” można napisać „metody”

Listing 1. Wzór dokumentacji funkcji

```
nazwa funkcji:      <nazwa>
opis funkcji:       <opis co robi funkcja>
parametry:          <nazwa i opis parametru1, lub „brak”>
                    <nazwa i opis parametru2>
zwracany typ i opis: <nazwa typu i opis co jest zwracane lub „brak”>
autor:              <numer zdającego>
```

Wykonaj zrzuty ekranu dokumentujące uruchomienie aplikacji utworzonych podczas egzaminu. Zrzuty powinny obejmować cały obszar ekranu monitora z widocznym paskiem zadań. Jeżeli aplikacja uruchamia się, na zrzucie należy umieścić okno z wynikiem działania programu oraz otwarte środowisko programistyczne z projektem lub okno terminala z kompilacją projektu. Jeżeli aplikacja nie uruchamia się z powodu błędów kompilacji, należy na zrzucie umieścić okno ze spisem błędów i widocznym otwartym środowiskiem programistycznym. Wykonać należy tyle zrzutów, ile interakcji podejmuje aplikacja. Wymagane zrzuty ekranu:

- Aplikacja konsolowa – dowolna liczba zrzutów nazwanych *konsola1*, *konsola2*, ...
- Aplikacja webowa – dowolna liczba zrzutów nazwanych *web1*, *web2*, ... (stan początkowy, interakcje z zaznaczonym i odznaczonym polem wyboru)

Utwórz plik tekstowy i nazwij go *egzamin*. Dokument powinien zawierać informacje o narzędziach wykorzystanych na egzaminie:

- Nazwę systemu operacyjnego
- Nazwy środowisk programistycznych
- Nazwy języków programowania

Zrzuty ekranu i dokument umieść w folderze *dokumentacja*.

UWAGA: Nagraj płytę z rezultatami pracy. W folderze z numerem zdającego powinny się znajdować trzy foldery: dokumentacja, konsolowa i webowa.

W folderze dokumentacja: pliki ze zrzutami oraz plik egzamin.

W folderze konsolowa: spakowany cały projekt aplikacji konsolowej, modyfikowane pliki źródłowe, plik wykonywalny, jeżeli istnieje.

W folderze webowa: spakowany cały projekt aplikacji webowej, modyfikowane pliki źródłowe.

Opisz nagraną płytę numerem zdającego i zapakuj w pudełko. Pozostaw ją na stanowisku wraz z arkuszem egzaminacyjnym.

Czas przeznaczony na wykonanie zadania wynosi 180 minut.

Ocenie będą podlegać 4 rezultaty:

- implementacja, kompilacja, uruchomienie programu,
- aplikacja konsolowa,
- aplikacja webowa,
- dokumentacja aplikacji.

Tabela 1. Wybrane elementy frameworka Angular, biblioteki React.js i biblioteki Bootstrap

Angular To use ngModel i ngForm add: <code>import { FormsModule } from '@angular/forms';</code> in <code>app.module.ts</code> file. Add <code>FormsModule</code> to imports table To use Bootstrap add to <code>styles.css</code> : <code>@import "~bootstrap/dist/css/bootstrap.css";</code>
React.js To use Bootstrap add: <code>import 'bootstrap/dist/css/bootstrap.css';</code>
Bootstrap forms Textual form controls—like <code><input>s</code> , <code><select>s</code> , and <code><textarea>s</code> —are styled with the <code>.form-control</code> class. <code><input type="email" class="form-control" id="exampleInputEmail1" /></code> Important! In React render method use <code>className</code> instead of <code>class</code>
Bootstrap buttons Bootstrap includes several predefined button styles, each serving its own semantic purpose, with a few extras thrown in for more control. The <code>btn</code> classes are designed to be used with the <code><button></code> element. Add modifier classes as: <code>btn-primary</code> , <code>btn-secondary</code> , <code>btn-success</code> to <code>btn</code> class in order to add background colors. e.g. <code><button type="button" class="btn btn-success">Success</button></code>
Bootstrap utilities – padding and margin, colors, paragraph formatting The classes are used in the format: <code>{property}-{size}</code> . Where property is one of: <code>m</code> - sets margin, <code>p</code> - sets padding Where size is one of: 0 - sets margin or padding to 0 1 - sets margin or padding to .25rem 2 - sets margin or padding to .5rem 3 - sets margin or padding to 1rem 4 - sets margin or padding to 1.5rem 5 - sets margin or padding to 3rem Bootstrap 5 has some contextual classes that can be used to provide "meaning through colors". The classes for text colors are e.g.: <code>.text-primary</code> , <code>.text-success</code> , <code>.text-danger</code> The Bootstrap 5 classes below can be added to style HTML elements: <code>.text-start</code> , <code>.text-nowrap</code> , <code>.lead</code> , <code>.text-capitalize</code>

Wypełnia zdający w przypadku nagranej płyty

**Do arkusza egzaminacyjnego dołączam płytę CD/DVD z rezultatami wykonania zadania egzaminacyjnego opisaną przeze mnie numerem PESEL

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

 i oświadczam, że jakość nagrania została przeze mnie sprawdzona.**

Wypełnia zdający w przypadku nienagranej (pustej) płyty

**Poświadczam numerem PESEL

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

, że do arkusza nie dołączam płyty CD/DVD z rezultatami wykonania zadania egzaminacyjnego.**

Wypełnia Przewodniczący ZN

- ☐ **Potwierdzam, że do arkusza egzaminacyjnego dołączona jest płyta CD/DVD, opisana numerem PESEL zdającego.**
- ☐ **Potwierdzam, że do arkusza egzaminacyjnego nie została dołączona płyta CD/DVD.**

.....
Czytelny podpis Przewodniczącego ZN