

Część I. Aplikacja Web

Wykonaj aplikację internetową typu front-end obsługującą wyświetlanie danych tekst i obrazek. Dane odnoszą się do znanej gry Pokemon. Twoim zdaniem jest utworzenie aplikacji internetowej z zastosowaniem frameworka Angular lub biblioteki React.

Dane

W katalogu ``data`` znajdują się następujące pliki:

- ``pokedex.json`` - lista wpisów, która składa się z nazwy oraz unikalnego ID
- ``images`` - pliki graficzne odpowiadające postacią Pokemon (nazwa składa się z ID oraz rozszerzenia).

Funkcjonalność aplikacji

Aplikacja składa się z wyświetlania kafli, które składają się z nazwy, obrazka oraz obramowania. Dodatkowym elementem aplikacji jest pasek, który ma wspomóc filtrowanie wszystkich danych. Za pomocą niego powinny wyświetlać się tylko takie elementy, które nazwy elementów zawierają wpisaną frazę w pole tekstowe. Filtrowanie nie powinno być wrażliwe na zmianę wielkości liter.

Graficzny styl aplikacji web

Kafel z Pokemonem powinien posiadać rozmiar 128px na 128px, jego obramówka składa się z grubości 2px, jest czarna oraz namalowana za pomocą kropek. Obrazek na kaflu ma znajdować się w jego centrum a jego rozmiar składa się z 75% (szerokość i wysokość). Tekst jest wyśrodkowany.

Kafle powinny znajdować się w kontenerze, który wyświetlany jest za pomocą flexbox, jego jedyną właściwością jest zawijanie się.

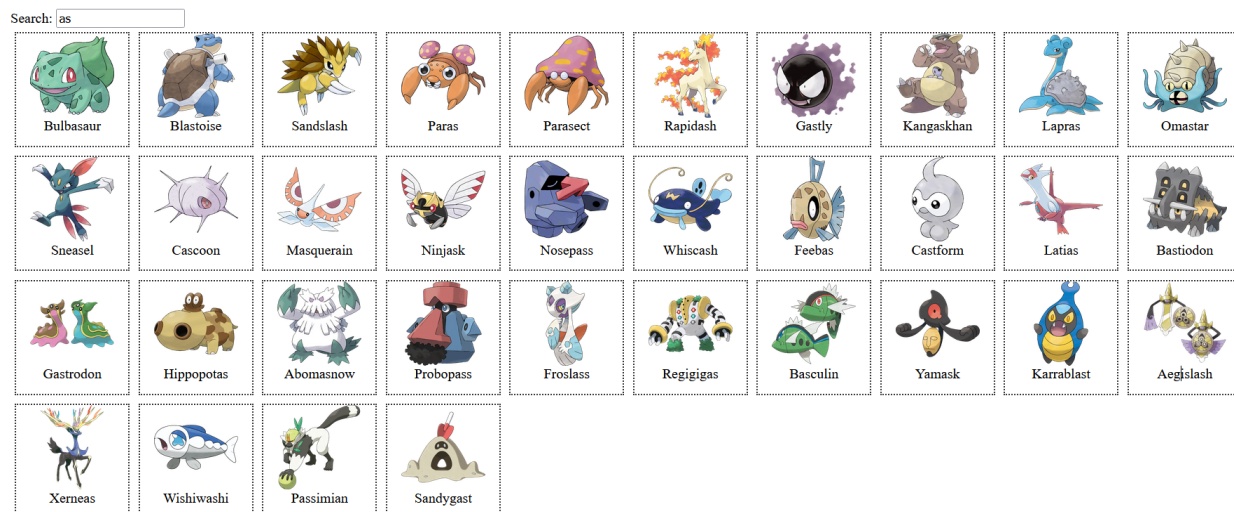
Filtrowanie

Element składa się z tekstu ``Search``, a bok niego znajduje się pole tekstowe do, którego można wprowadzić tekst. Celem filtrowania jest pominięcie elementów, które nie zawierają w nazwie elementu wpisanej frazy w polu tekstowym. Filtrowanie nie powinno być wrażliwe na zmianę wielkości liter.

Podgląd (ogólny wygląd)



Podgląd (filtrowanie)



Wybrane elementy frameworka Angular, biblioteki React.js

React.JS	
Wczytanie pliku json	<code>import variable from `./file.json`</code>
Wczytanie obrazka dynamicznego	<code>src={require('./images/filename.png')}</code>

Angular	
Wczytanie pliku json	<code>Import variable from `./file.json`</code>
Wczytanie obrazka dynamicznego	<code>src="./images/filename.png"</code>

Część II. Aplikacja konsolowa

Korzystając z mechanizmów programowania obiektowego zaprojektuj część logiki systemu forum użytkowników. Zaimplementuj klasę o nazwie osoba z konstruktorami, metodą i obsługą pola statycznego.

Założenia klasy:

- Zastosowano obiektowy język programowania zgodny z zainstalowanym na stanowisku egzaminacyjnym: C++ lub C#, lub Java, lub Python.
- Klasa Osoba zawiera:
 - Dwa pola: numeryczne id oraz tekstowe imię osoby. Dostęp do obu pól ma jedynie klasa. W przypadku późniejszego rozszerzenia klasy, klasy potomne nie mają dostępu do tych pól.
 - Ogólnie dostępne pole statyczne zliczające liczbę instancji klasy, początkowo wypełnione wartością 0.
 - Trzy konstruktory:
 - bezparametrowy, ustawia wartości 0 i pusty dla pól,
 - z dwoma parametrami, które przekazują wartości id i imienia,
 - kopiujący.
 - Każdy z konstruktorów dodatkowo inkrementuje liczbę instancji klasy.
 - Metodę do wypisania imienia obiektu klasy Osoba w postaci: „Cześć <argument>, mam na imię <imie>”, gdzie pole <argument> jest innym imieniem przekazany jako parametr wejściowy metody. Jeżeli w obiekcie nie wypełniono wartości pola z imieniem wyświetlany jest komunikat „Brak danych”.

Uwaga: W języku Python, zamiast przeciążenia konstruktorów, należy utworzyć konstruktor z dwoma parametrami o wartościach domyślnych (0, pusty napis), który będzie zastępował konstruktor bezparametrowy oraz konstruktor z dwoma parametrami. Następnie zaimplementować metodę kopiującą dane z jednego obiektu do drugiego.

- Program powinien być zapisany czytelnie, z zachowaniem zasad czystego formatowania kodu, należy stosować znaczące nazwy pól i metod.
- Do klasy należy dołączyć dokumentację oraz testy w postaci programu głównego. Testy zostały opisane w części 4 zadania egzaminacyjnego.

Kod aplikacji przygotuj do nagrania. W podfolderze konsola powinno znaleźć się archiwum całego projektu o nazwie konsola.zip, plik z kodem źródłowym klasy oraz plik uruchomieniowy, jeżeli istnieje.

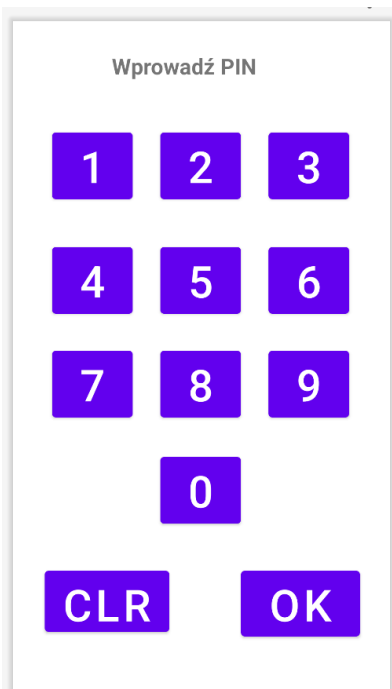
Część III. Aplikacja mobilna

Wykonaj aplikację mobilną za pomocą środowiska programistycznego dostępnego na stanowisku egzaminacyjnym oraz uruchom ją w dostępnym emulatorze systemu mobilnego. Aplikacja powinna składać się z dwóch Activity.

W pierwszym Activity wprowadzamy numer PIN i możemy go zatwierdzić lub wyczyścić to co wprowadziliśmy. W drugim Activity pojawia się wprowadzony numer PIN i wciskając przycisk „POWRÓT” możemy wrócić do pierwszego Activity.

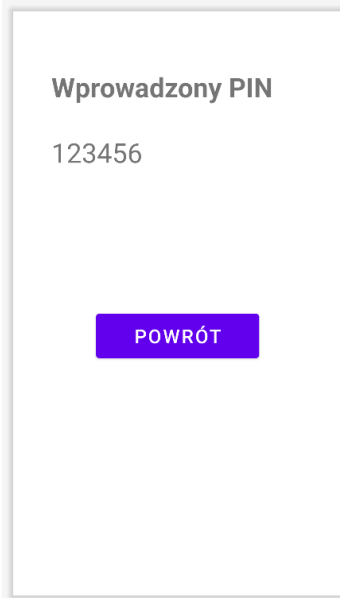
Elementy pierwszego Activity:

- Tytuł o treści: „Wprowadź PIN”.
- Dziesięć przycisków o treści: od „0” do „9” umiejscowione obok siebie maksymalnie po trzy w rzędzie jak na załączonym poniżej obrazku
- Dwa przyciski o treści: „CLR” do „OK” umiejscowione obok siebie w najniższym rzędzie jak na załączonym poniżej obrazku.



Elementy drugiego Activity:

- Tytuł o treści: „Wprowadzony PIN”
- Tytuł o treści zgodnym z wprowadzonym w pierwszym Activity numerem PIN
- Przycisk o treści: „POWRÓT” powodujący powrót do pierwszego Activity
- Elementy umiejscowione jeden pod drugim jak na załączonym poniżej obrazku.



Po wprowadzeniu numeru i zatwierdzeniu go przyciskiem „OK” w drugim Activity pojawia się wprowadzony numer PIN. Wygląd drugiego Activity znajduje się poniżej:

Działanie aplikacji:

- Aplikacja implementuje klawiaturę Pad’a numerycznego do wprowadzania numeru PIN.
- Przycisk „CLR” powoduje wyczyszczenie wprowadzonego numeru PIN
- Przycisk „OK” zatwierdza wprowadzony numer PIN
- Poprawna długość numeru PIN to cztery do sześciu cyfr. Jeśli użytkownik wprowadzi mniej cyfr niż cztery, to przycisk OK nie jest aktywny. Po podaniu sześciu cyfr automatycznie pokazuje się drugie Activity z wypisanym wprowadzonym numerem PIN

1. Scenariusz działania aplikacji:

wciskamy następujące przyciski: 1 2 3 4 5 0 program otwiera drugie activity w którym pojawia się informacja, że wprowadzony numer PIN to 123450

2. Scenariusz działania aplikacji:

wciskamy następujące przyciski: 4 2 3 4 5 OK program otwiera drugie activity w którym pojawia się informacja, że wprowadzony numer PIN to 42345

3. Scenariusz działania aplikacji:

wciskamy następujące przyciski: 4 2 CLR 1 2 5 6 OK program otwiera drugie activity w którym pojawia się informacja, że wprowadzony numer PIN to 1256

Po powrocie z drugiego Activity do pierwszego. Wprowadzanie numery PIN rozpoczynamy od nowa.

Założenia aplikacji:

- Interfejs użytkownika zapisany za pomocą języka znaczników wspieranego w danym środowisku (np. XAML, XML).
- Czcionka na przyciskach powinna być możliwie duża i taka aby wyświetlany tekst mieścił się na przycisku.
- Aplikacja powinna być zapisana czytelnie, z zachowaniem zasad czystego formatowania kodu, należy stosować znaczące nazwy zmiennych i funkcji.

Podjmij próbę kompilacji i emulacji. Informacje dotyczące dokumentacji i zrzutów ekranowych umieszczono w części IV zadania.

Kod aplikacji przygotuj do nagrania na płytę. W podfolderze *mobilna* powinno znaleźć się archiwum całego projektu o nazwie *mobilna.zip*

Część IV. Dokumentacja aplikacji

Wykonaj testy aplikacji konsolowej oraz dokumentację do aplikacji utworzonych na egzaminie. W podfolderze konsola w programie głównym aplikacji konsolowej należy sprawdzić działanie klasy poprzez, kolejno:

- Wyświetlenie komunikatu „Liczba zarejestrowanych osób to <licznik>”, gdzie <licznik> jest wartością pobraną z pola statycznego klasy.
- Utworzenie obiektu za pomocą konstruktora bezparametrowego.
- Utworzenie obiektu za pomocą konstruktora z dwoma parametrami. Dane obiektu wprowadzane z klawiatury.
- Utworzenie obiektu za pomocą konstruktora kopiującego (w Python konstruktora bezparametrowego i metody kopiującej). Obiekt z wypełnionymi polami jest źródłem kopiowania danych.
- Wywołanie metody do wypisania imienia z parametrem wejściowym równym „Jan” dla wszystkich utworzonych obiektów.
- Ponowne wyświetlenie komunikatu „Liczba zarejestrowanych osób to <licznik>”, gdzie <licznik> jest wartością pobraną z pola statycznego klasy

Wykonaj dokumentację do aplikacji utworzonych na egzaminie. W kodzie źródłowym aplikacji konsolowej utwórz nagłówek funkcji przeszukującej, według wzoru zgodnie z listingiem 1. Nagłówek powinien znaleźć się w kodzie źródłowym nad funkcją. W miejscu znaków <> należy podać nazwę funkcji, nazwy argumentów wraz z krótkim opisem, typ zwracany wraz z krótkim opisem, krótki opis zawierający przynajmniej nazwę algorytmu. Gdy funkcja nie ma argumentów – zapisać „brak”. W miejscu autor należy podać swoje nazwisko i imię.

UWAGA: Dokumentację 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

Listing 1. Wzór dokumentacji funkcji

```
*****
nazwa funkcji: <tu wstaw nazwę funkcji>
argumenty: <nazwa argumentu> - <co przechowuje>
           <nazwa argumentu> - <co przechowuje>
typ zwracany: <nazwa typu>, <co jest zwracane>
informacje: <opis>
autor: <Nazwisko i imię 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 internetowa – dowolna liczba zrzutów nazwanych *internetowa1*, *internetowa2* ...
- Aplikacja konsolowa – dowolna liczba zrzutów nazwanych *konsolowa1*, *konsolowa2* ...
- Aplikacja mobilna – dowolna liczba zrzutów nazwanych *mobile1*, *mobile2* ...

W edytorze tekstu pakietu biurowego utwórz plik z dokumentacją i nazwij go *egzamin*. Dokument powinien zawierać podpisane zrzuty ekranu oraz zapisane informacje:

- Nazwę systemu operacyjnego, na którym pracował zdający.
- Nazwy środowisk programistycznych, z których zdający korzystał na egzaminie.
- Nazwy języków programowania.
- Nazwa emulatora systemu mobilnego.

Zrzuty ekranu i dokument umieść w podfolderze *dokumentacja*.

Nagraj folder z rezultatami pracy. W folderze z Nazwiskiem i imieniem zdającego powinny się znajdować podfoldery: internetowa, konsolowa, mobilna, dokumentacja.

W folderze dokumentacja: pliki ze zrzutami oraz plik egzamin.

W folderze internetowa: spakowany cały projekt aplikacji internetowej, część serwerowa oraz klientowa

W folderze konsolowa: spakowany cały projekt aplikacji konsolowej, pliki z kodem źródłowym, plik uruchomieniowy

W folderze mobilna: spakowany cały projekt aplikacji mobilnej, pliki ze źródłami interfejsu i logiki.

Upewnij się, że wszystko co stworzyłeś zostało prawidłowo nagrane