

[pawel.rajba@gmail.com](mailto:pawel.rajba@gmail.com), <http://itcourses.eu/>

# Google Maps

# Plan wykładu

- Wprowadzenie
- Zasoby w sieci
- Zaczynamy z Google Maps
- Mapa
- Obiekty na mapie
  - Markery
  - Ikony
  - Okienka
  - Przegląd dalszych przykładów
- Podstawy zdarzeń
- Geocoding
- Places Search

# Wprowadzenie

- Jakie możliwości mają usługi map?
  - Wskazywanie i wyszukiwanie lokalizacji obiektów
  - Wskazywanie obiektów w pobliżu lokalizacji
  - Wyznaczanie trasy pomiędzy punktami
    - również z punktami pośrednimi
    - możliwość wydrukowania instrukcji dla kierowcy
  - Łączenie aktywności z lokalizacjami (np. fotki, filmy)
- Najpopularniejsze produkty
  - GoogleMaps
  - OpenStreetMaps
- Nas będzie interesować od strony programisty

# Wprowadzenie

- W dalszej części przyjrzymy produktom Google
- Dwie główne usługi
  - Google Maps
  - Google Places
- Dostępne
  - W postaci API, jako web services
  - Dla web, iOS i Android
- Do pewnych limitów za darmo, powyżej płatne
  - Nie ma już potrzeby generować kluczy dla domen
  - Klucze są potrzebne do wykorzystania komercyjnego
    - Zarządzanie usługami: <https://console.developers.google.com/>

# Zasoby w sieci

- Strona główna Google Maps API i Places API
  - <https://developers.google.com/maps/>
  - <https://developers.google.com/places/>
- Dokumentacja API w wersji 3
  - <https://developers.google.com/maps/documentation/javascript/>
  - <https://developers.google.com/places/documentation/>
- Kursy
  - <http://gmapsapi.com/>
  - <http://w3schools.com/googleAPI/>
- Przykłady
  - <http://econym.org.uk/gmap/>
- Ciekawe artykuły
  - <http://grafmag.pl/artykuly/mapa-na-stronie-internetowej-przy-wykorzystaniu-google-maps-api/>
  - <http://googlegeodevelopers.blogspot.co.uk/2012/05/google-places-api-search-refinements-as.html>

# Zasoby w sieci

- Wybrane ciekawe usługi
  - Places Search  
<https://developers.google.com/places/documentation/search>
  - Places Autocomplete  
<https://developers.google.com/places/documentation/autocomplete>
  - Directions  
<https://developers.google.com/maps/documentation/javascript/directions>
  - Distance Matrix  
<https://developers.google.com/maps/documentation/distancematrix>
  - Geocoding  
<https://developers.google.com/maps/documentation/javascript/geocoding>
  - Street View  
<https://developers.google.com/maps/documentation/javascript/streetview>

# Zaczynamy z Google Maps



- Źródło: [http://gmapsapi.com/images/wstep\\_v3.jpg](http://gmapsapi.com/images/wstep_v3.jpg)

# Zaczynamy z Google Maps

- Do budowania aplikacji mapowych wykorzystujemy obiekty `google.maps.*`
- Przekazywanie opcji realizujemy przez obiekty w notacji JSON
  - Ogólna składnia:
    - `var opcje = { opcja: wartosc, ... }`
  - Przykład:
    - `var mapOptions = { zoom: 10, center: coords, mapTypeId: google.maps.MapTypeId.ROADMAP };`



# Zaczynamy z Google Maps

- W pliku z mapą wprowadzamy
  - Wstawienie biblioteki Google Maps API
    - Obowiązkowe ustawienie opcji geolokalizacyjnej sensor
  - DIVa na mapę
  - Kod inicjujący mapę

# Mapa

- Utworzenie mapy
  - `new google.maps.Map(  
document.getElementById("map"),  
mapOptions);`
- Lista opcji
  - `backgroundColor, center, draggable, mapTypeId, zoom`
- `MapTypeId` to jedna z wartości `google.maps.MapTypeId.X`, gdzie X to:
  - `HYBRID, ROADMAP, SATELLITE, TERRAIN`

# Mapa

- Dodatkowe opcje
  - `mapTypeControl` – określa rodzaj mapy
    - Lista dostępnych typów: `mapTypeId`  
HYBRID, ROADMAP, SATELLITE, TERRAIN
    - Sposób wyświetlania: `style`  
DROPDOWN\_MENU, HORIZONTAL\_BAR
  - `scaleControl` – miarka skali
  - `navigationControl` – określa panel powiększania
    - Sposób wyświetlania: `style`  
ANDROID, ZOOM\_PAN, SMALL

# Przykład

---

- SimpleStart

# Obiekty na mapie

- Obiektami na mapie nazywamy tzw. overlays
- Na mapie możemy osadzić
  - Marker
  - Icon
  - InfoWindow
  - Polyline
  - Rectangle
  - Circle
  - Polygon
  - ... i kilka innych

# Markery

- Utworzenie markera
  - `new google.maps.Marker(opcje)`
- Lista opcji
  - `position, map, draggable, clickable, icon, visible, shadow, zIndex`
- Marker może mieć dowolną ikonę
  - Określamy przez `icon`, która jest typu
  - `google.maps.MarkerImage`

# Ikony markerów

- Utworzenie ikony
  - `new google.maps.MarkerImage(adresobrazka, rozmiar, punkt_startowy, punkt_zaczeplenia);`
- Rozmiar jest typu
  - `google.maps.Size`
- Punkty są typu
  - `google.maps.Point`
- Darmowe ikony są zebrane pod adresem
  - [http://gmapsapi.com/poradnik/103\\_popularne\\_darmowe\\_ikony\\_dla\\_markerow.html](http://gmapsapi.com/poradnik/103_popularne_darmowe_ikony_dla_markerow.html)

# Przykłady

---

- Markers
- MarkersIcons



# Okienka

- Okienka czyli inaczej dymki
  - Można do nich wstawić prawie dowolną treść
- Utworzenie
  - `new google.maps.InfoWindow();`
- Co możemy zrobić?
  - `infoWindow.setContent('Tekst z informacją');`
  - `infoWindow.setPosition(  
    new google.maps.LatLng(lat, long));`
  - `infoWindow.open(map[, marker]);`

# Przykład

---

- InfoWindow

# Dalsze przykłady overlays

- Patrzymy na przykłady:

[http://w3schools.com/googleAPI/google\\_maps\\_overlays.asp](http://w3schools.com/googleAPI/google_maps_overlays.asp)

# Podstawy zdarzeń

- Do obsługi zdarzeń jest funkcja
  - `google.maps.event.addListener()`, parametry
    - Obiekt mapy
    - Rodzaj zdarzenia (string)
    - Funkcja obsługi zdarzenia
      - jej parametrem jest obiekt zdarzenia
  - Przykładowe zdarzenia
    - `click`, `dblclick`, `mouseup`, `mousedown`, `mouseover`, `mouseout`, `zoom_changed`

# Przykład

---

- Events

# Geocoding

- Pozwala na wyszukaniu współrzędnych na podstawie adresu
- Przykład
  - Geocoders

# Places Search

- Pozwala na wyszukiwanie obiektów w pobliżu
- Przykłady
  - <https://developers.google.com/maps/documentation/javascript/examples/place-search>
  - <https://developers.google.com/maps/documentation/javascript/examples/places-searchbox>