

Radionica 3

Ponavljjanje

- Organizacija projekta
- SettingsViewController - UITableView sa statičkim ćelijama
- Singleton
- NSUserDefaults
- CLLocationManager

Današnji plan

- Custom table view cell
- UITableView s dinamičkim sadržajem
- Dohvaćanje i obrada podataka
- Prikaz podataka na ekranu

Custom UITableView ćelije

- UITableViewCell subclass
- Dodavanje UI elemenata i povezivanje sa klasom ćelije
 - Autolayout



UITableViewCellDelegate

UITableViewCellDataSource

- (NSInteger)numberOfSectionsInTableView:(UITableView *)tableView;
- (NSInteger)tableView:(UITableView *)tableView
 numberOfRowsInSection:(NSInteger)section;
- (UITableViewCell *)tableView:(UITableView *)tableView
 cellForRowAtIndexPath:(NSIndexPath *)indexPath;

“Template” implementacije

– (UITableViewCell *)tableView...

```
– (UITableViewCell *)tableView:(UITableView *)tableView
  cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:@"reuseID"
                                                                    forIndexPath:indexPath];

    // Configure the cell, for example:
    // cell.textLabel.text = @"Text";

    return cell;
}
```

Načini dohvaćanja podataka putem Interneta

- NSURLConnection
- NSURLSession
 - novije i naprednije od NSURLConnection
- Nešto treće?

NSURLSession primjer

```
NSURLSession *session = [NSURLSession sharedSession];
NSURL *url = [NSURL URLWithString:@"http://echo.jsontest.com/key/value/one/two"];

NSURLSessionDataTask *task = [session dataTaskWithURL:url
    completionHandler:^(NSData *data, NSURLResponse *response, NSError *error) {
        // handle response and data
    }];
[task resume]; // start task
```

- completionHandler - blok
 - kao argument metodi dajemo dio koda koji će se izvršiti kad HTTP zahtjev završi
 - pozvat će se s argumentima data, response i error

Xcode primjer

JSON

- JavaScript Object Notation
- format zapisa lako čitljiv ljudima primarno korišten za prijenos podataka putem Interneta
- tipovi podataka:
 - number, string, boolean, array, object, null

JSON primjer

```
{
  "number": 1.23,
  "string": "text",
  "bool": false,
  "object": {
    "key": "value"
  },
  "array": [
    1,
    2,
    3
  ],
  "nothing": null
}
```

NSJSONSerialization

```
+ (NSData *)dataWithJSONObject:(id)obj
                             options:(NSJSONWritingOptions)opt
                             error:(NSError **)error;

+ (id)JSONObjectWithData:(NSData *)data
                      options:(NSJSONReadingOptions)opt
                      error:(NSError **)error;
```

Xcode primjer

Kako ćemo mi to napraviti?

- iskoristi ćemo neke gotove projekte koji olakšavaju cijeli proces
- JSONModel:
 - <https://github.com/icanzilb/JSONModel>
- AFNetworking:
 - <https://github.com/AFNetworking/AFNetworking>

JSONModel

- Automatski popunjava Objective-C model iz JSON podataka
- potrebno je definirati property-e i odrediti kako će se podaci mapirati u njih iz JSON-a
- primjer korištenja:
 - README datoteka u repozitoriju

AFNetworking

- najpopularniji Objective-C repozitorij na Githubu
 - izvor: [Github statistika za Objective-C prema broju zvijezdica](#)
- primjer korištenja:
 - README datoteka u repozitoriju

Cocoapods

“CocoaPods is the dependency manager for Objective-C projects.”

- Podfile:
`$ pod init`
- Novi “dependency”:
`pod 'AFNetworking'`
- Install:
`$ pod install`

Weather API

- <https://developer.worldweatheronline.com>
- <https://api.worldweatheronline.com/free/v2/weather.ashx>
- parametri:

key: 3f9356ccea3ad483718ca9fed95dd0

format: json

num_of_days: 6

q: lokacija ili latitude/longitude

Xcode!

Prikaz podataka

- osvježavanje podataka na ekranu
 - osvježavanje tablice s desne strane
- prosljeđivanje podataka controlleru za prikaz trenutnog vremena i osvježavanje UI elemenata
- slike vremena - implementirati dinamički readonly property u modelu