# Mobile Development: Plan for week 06 [week 05](M2016F_plan_Week05.pdf) - [week 08](M2016F_plan_Week08.pdf)

## Goal for this week

* You should have knowledge about way of making data persistent and save some simple data.
* You should be able to use Shared Preferences and Android file system for keeping data persistent including using java serialization.
* You should be able to make simple exchange of data with a website / webservice in your app – using synchronize HTTP request (using Android 2.2 – API 8) – next week async

**Preparation before Tuesday 9th February**

* viewing following toturial this playlist: <https://www.youtube.com/playlist?list=PL6gx4Cwl9DGBsvRxJJOzG4r4k_zLKrnxl> **65, 66 (Shared Preferences)**
* **reading briefly about** 
  + Tutorial - saving data:<https://developer.android.com/training/basics/data-storage/index.html>
  + Tutorial - Connecting to the Network: <http://developer.android.com/training/basics/network-ops/connecting.html>  
    Reference – HttpURLConnection class: <http://developer.android.com/reference/java/net/HttpURLConnection.html>

## Teachers plan for Tuesday 9th February (9:00-12:00)

* Follow up on the weeks of study (adjusting plan for week together)
* Persistence (files and shared preferences) inclusive serialization  
  slides: [Android\_Persistens.ppt](http://bjoerks.net/klasser/Mobile_2015_Foraar/materials/Android_Persistens.ppt)  
  examples: [Android\_AppFiles\_20150225.zip](http://bjoerks.net/klasser/Mobile_2015_Foraar/materials/Android_AppFiles_20150225.zip)
* Accessing webservice for external data  
  I use HttpRLConnection in my example and not HttpClient, as I discovered there is discussion about only HttpURLConnection is supported by the Android team.  
  For these examples you will have to choose an and old Android version fx 2.2.3 as you now only are allowed to use asynchronous access for demand of liveness – but for the beginning you might find it easier to debug when using synchronous call.  
  [Android\_HttpGetRequest\_Syncrone\_20150304.zip](http://bjoerks.net/klasser/Mobile_2015_Foraar/materials/Android_HttpGetRequest_Syncrone_20150304.zip) (Basic of making synchronized web call - http get request – only allowed in older android versions)  
  [Android\_HttpGetPostRequest\_Syncrone\_20150304.zip](http://bjoerks.net/klasser/Mobile_2015_Foraar/materials/Android_HttpGetPostRequest_Syncrone_20150304.zip) (Basic of making synchronized web call - http post request – only allowed in older android versions)  
  [Android\_HttpGetPostRequest\_AssertXmlReq\_20150304.zip](http://bjoerks.net/klasser/Mobile_2015_Foraar/materials/Android_HttpGetPostRequest_AssertXmlReq_20150304.zip) (Expanded example with trick to make soap web call a little more easy).  
  Next week will be same example/template expanded to be running asynchronous:

## Extra useful tools for making interacting with webservices:

* + With WCF services (and ASMX) you can get the necessary data for SOAP call with WcfTestClient.exe you probably will find here "C: \ Program Files (x86) \ Microsoft Visual Studio ….. \ …… \ IDE \ WcfTestClient.exe"
  + Fidler is a usefull tool to show all what is sent and received on the http protocol: <http://www.telerik.com/download/fiddler>  
    You can use this tool for discover how to interact with a webservice, by using the service from other applications.

## Preparation before Friday 12th February

* Expand your apps from week 05 with using Shared Preferences or file system for making information as mail / phone numbers for parent ect. Persistent. (and debug them on device)
* Make one or to apps that can retrieve data from webservice or send data to webservice for update  
  Ideas for small apps/exercises: USE API 8 to allow synchronous access
  + External data to be used with web access
    - Webservices for retrieving data <http://webservicedemo.datamatiker-skolen.dk/>   
      Both with soap and REST call for JSON and XML
    - Webservices for retrieving data <http://wcfdemo.datamatiker-skolen.dk/>
    - Webservice with geodata from Odense Municipality <http://geodata.odense.dk/datatothepeople/> for retrieving JSON or XML data for fx parks and playgrounds.
  + Make an app for simple calculation (add and subtract), which use a webservice for doing the calculation it self.
  + Make an app, which can show a list of persons retrieved from webservice  
    When selecting an item show detail in another screen (activity)
  + Make an app, which can show a list of data retrieved from webservice – fx list of parks in Odense  
    When selecting an item show detail in another screen (activity)

## Teachers plan for Friday 12th February – no lecture

* All teachers is on a meeting this Friday, so you should work on your own.
* Together in small groups make a walk through your apps accessing webservices for getting and posting data.