

# *Web services REST in NetBeans con Spring Boot*

## *Installazione del plugin*

Installare il plugin "NB SpringBoot" o dal portale (<http://plugins.netbeans.org/plugin/67888/nb-springboot>) o da **Tools / Plugins** (una volta installato verrà listato nella scheda Installed tra gli User Installed Plugins).

## *Creazione di un progetto*

New project / Categories: Maven / Projects: Spring Boot Initializr project

Base properties:

- Group: è la società, per rendere univoco il nome completo (es. edu.alex)
- Artifact: è il nome del progetto (es. demo)
- Packaging: jar o war

Dependencies:

- Core: Dev tools
- Web: Web

Name & location: specificare il nome del progetto e la directory dove salvarlo.

## *Packaging*

Se il packaging è Jar non ci sono problemi: viene avviato un server Tomcat interno che permette di eseguire l'applicazione; il server si mette in ascolto sulla porta 8080 (che DEVE essere libera).

Se il packaging è War bisogna usare un server compatibile con Java 8 o succ. (e comunque controllare anche altre specifiche): se si usa Tomcat, questo deve essere di versione 9 o succ.

## *Creazione del codice*

Click dx nel nodo Source packages / [Group.Artifact] / New Java class creare una classe controller con l'mapping per le richieste al web service, ad es.:

```
@RestController
public class MioController {

    @RequestMapping("/")
    public String index() {
        return "Hello!";
    }
}
```

## *Esecuzione del progetto*

Se si è scelto il packaging Jar, premere il pulsante Run project [F6], aprire un browser e portarsi all'url <http://localhost:8080/>; dovrebbe apparire la "pagina" mappata sulla radice (ovviamente la porta 8080 non doveva essere occupata in precedenza, altrimenti l'embedded server Tomcat non parte).

Se si cambia il codice basta salvare il file. La dipendenza "Devtools" carica automaticamente nel web server la nuova versione (deploy automatico): per vedere il sito aggiornato basta aggiornare la pagina del browser ([F5]). [NB: la prima volta non funziona, poi sì]

Se si è scelto il packaging War si deve fare manualmente il deployment del file War sull'application server desiderato, che deve essere conforme alle specifiche Java (per Tomcat vedi <http://tomcat.apache.org/whichversion.html>)