

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشکده مهندسی کامپیوتر

راهنمای شروع کار با جاوا

تهیه کننده:

سید احسان موسی کاظمی محمدی

دانشجوی کارشناسی ارشد

مهندسی فناوری اطلاعات

email: mousakazemi@stu.yazduni.ac.ir

زمستان ۸۸

فهرست مطالب

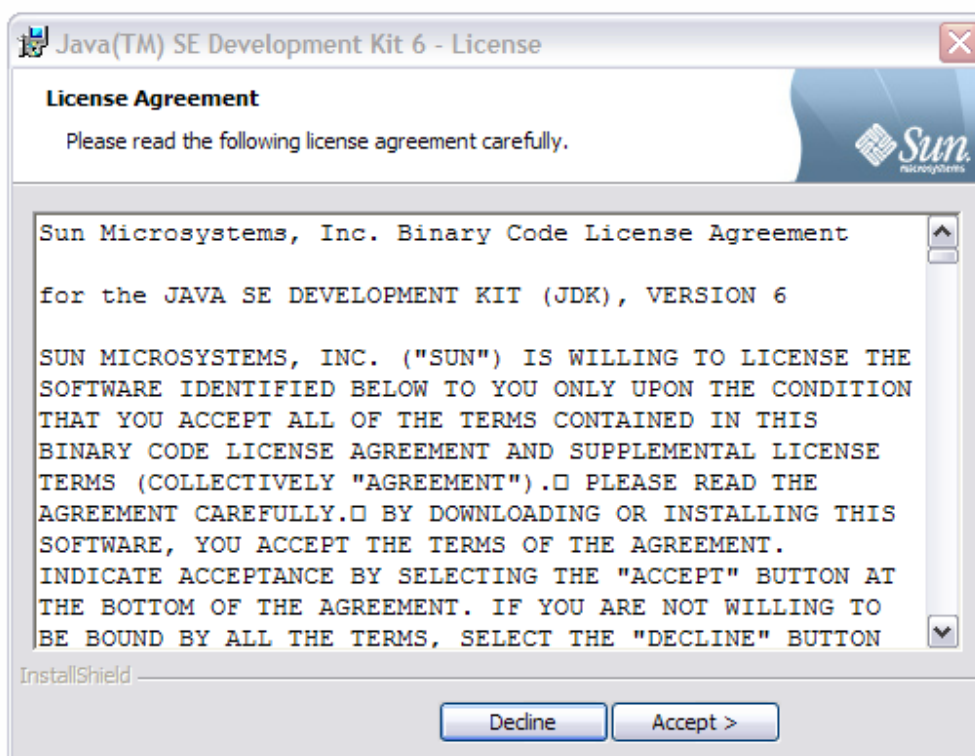
۴(۱) نصب JDK و JRE
۸(۲-۱) تنظیم متغیرهای محیطی
۱۲(۲) نصب IntelliJ IDEA و ایجاد اولین پروژه
۳۹(۳) کامپایل و اجرای برنامه‌ها از خط Command
۴۱(۴) نکات متفرقه

(۱) نصب JDK و JRE

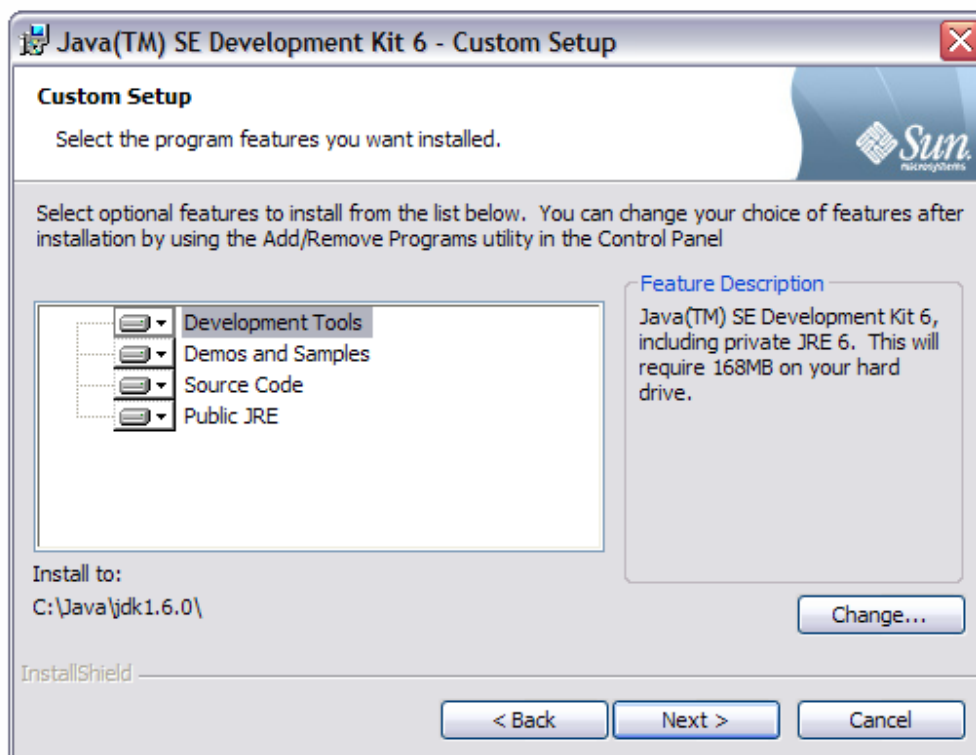
در ابتدا لازم است تا JDK نصب شود. JDK مخفف Java Development Kit بوده و محصولی است که شرکت Sun Microsystems در راستای کمک به برنامه نویسان جاوا جهت ایجاد برنامه‌هایشان، در اختیار آنها قرار داده است. از جمله ابزارهای اصلی در JDK می‌توان به کامپایلر جاوا (javac)، ماشین مجازی جاوا (java) و jar ساز (jar) اشاره نمود.

نسخه ای از JDK که در این راهنما در نظر گرفته شده، نسخه ۶ ام آن است.

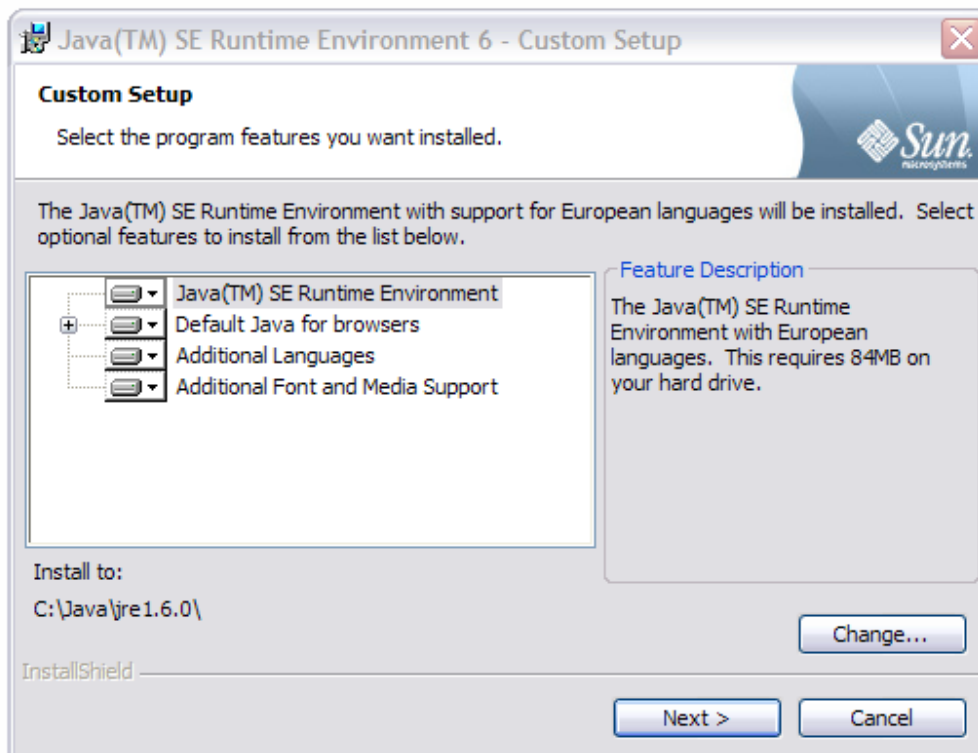
- ۱- برای شروع عملیات نصب بر روی فایل jdk-6-windows-i586 دابل کلیک کنید تا پنجره زیر باز شود. بر روی دکمه > Accept کلیک کنید.



۲- در این صفحه می‌توانید با کلیک بر روی **Change** مسیر پیش فرض نصب را تغییر دهید. برای مثال من تصمیم دارم تا آن را در مسیر `C:\Java\jdk1.6.0` نصب کنم. شما می‌توانید همان مسیر پیش فرض یعنی `C:\Program Files\Java\jdk1.6.0` را بپذیرید و آن را تغییر ندهید. بر روی دکمه **Next >** کلیک کنید.



۳- بعد از چند ثانیه (بسته به مشخصات سیستم‌تان)، پنجره زیر باز می‌شود که در آن می‌توانید مسیر نصب JRE را تعیین کنید. (JRE مخفف Java Runtime Environment است.) من تصمیم دارم آن را در مسیر C:\Java\jre1.6.0 نصب کنم. شما می‌توانید مسیر پیش فرض را پذیرفته و آن را تغییر ندهید.

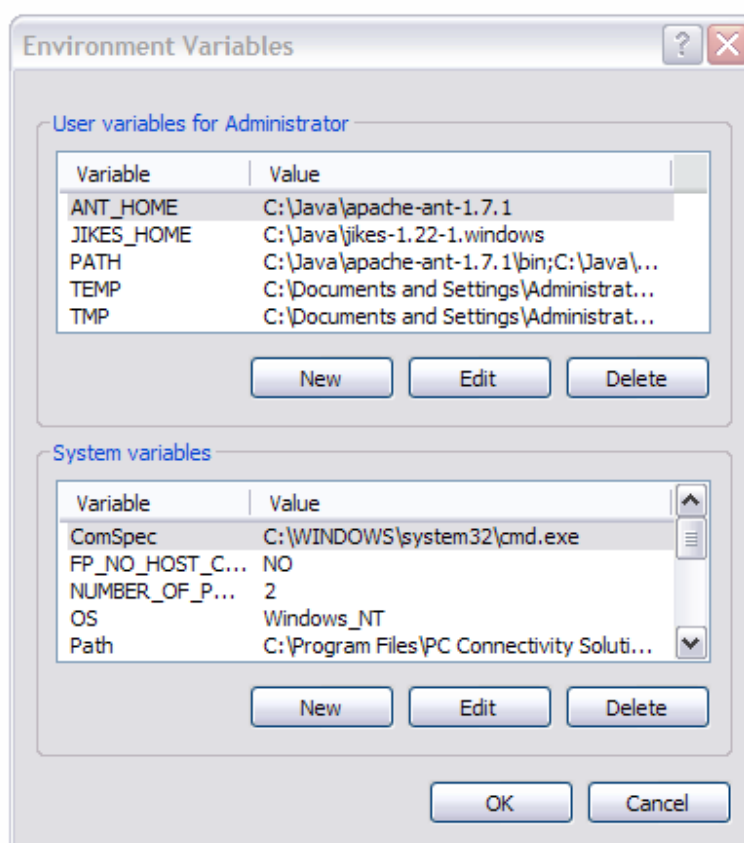


۴- در انتها پنجره زیر نمایش داده می‌شود. اگر مانند من حوصله خواندن فایل readme را ندارید، checkbox مربوط به Show the readme file را از حالت انتخاب خارج کنید. بر روی Finish کلیک کنید.

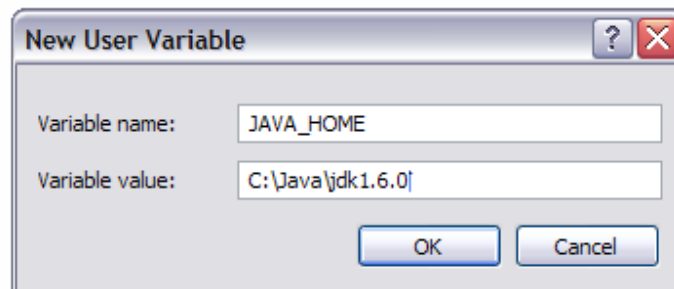


۲-۱) تنظیم متغیرهای محیطی

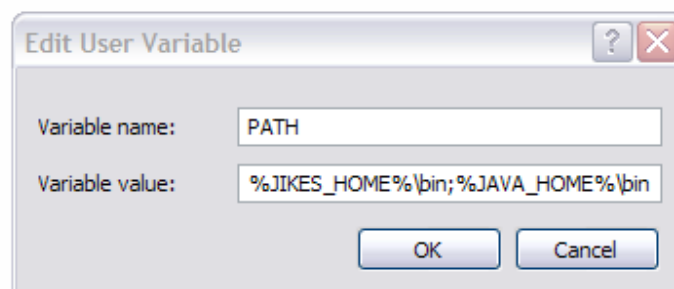
در ادامه لازم است تا متغیرهای محیطی (Environment Variables) لازم را تنظیم کنیم. بدین منظور بر روی My Computer کلیک راست نموده و با ادامه دادن مسیر Environment Variables -> Advanced -> Properties به پنجره متغیرهای محیطی برسید. در این پنجره قصد داریم تا متغیر JAVA_HOME را تنظیم کنیم. می‌توانیم آن را در بخش User variables یا System variables ایجاد کنیم. من آن را در بخش User variables ایجاد می‌کنم.



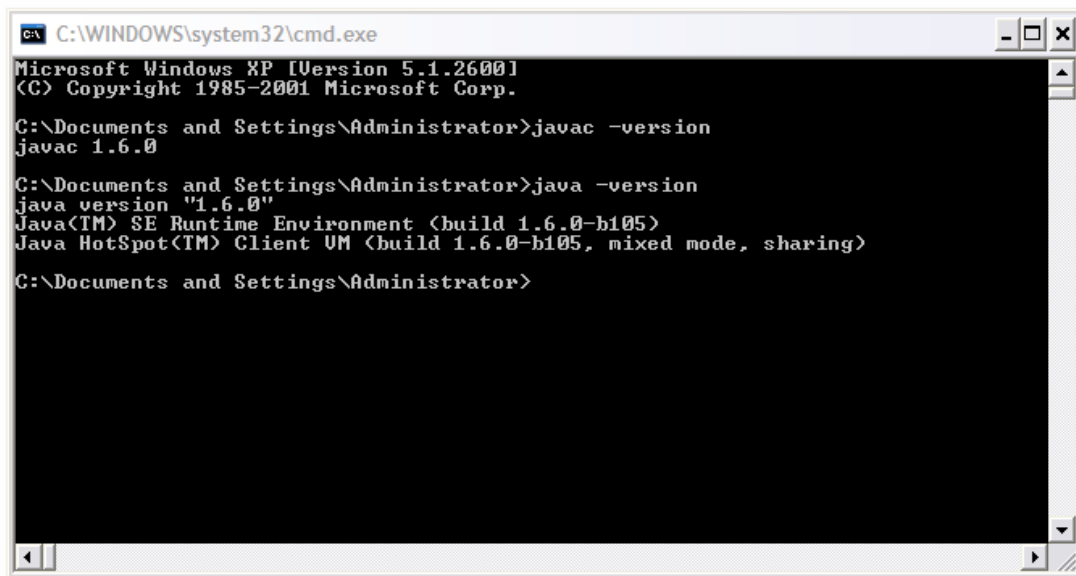
بدین منظور بر روی دکمه New کلیک نموده و در بخش Variable name کلمه JAVA_HOME و در بخش Variable value مسیر نصب JDK را وارد کنید.



حال در همان بخش User variables (نه System variables) به دنبال متغیر PATH بگردید. (اگر وجود ندارد، آن را ایجاد کنید. بر روی New کلیک نموده و Variable name را PATH قرار دهید.) در قسمت Variable value عبارت %JAVA_HOME%\bin را وارد کنید. (اگر PATH قبلاً موجود بوده و دارای مقدار نیز است، می‌بایست در انتهای مقدار آن ؛ قرار داده و عبارت فوق را به انتهای آن اضافه کنید.) در انتها تمام پنجره‌های باز را OK کنید. (با بقیه مقادیری که در تصویر زیر می‌بینید، مانند JIKES_HOME و غیره، کاری نداشته باشید.)



برای اطمینان از نصب Java می‌توانید وارد محیط Command Prompt شده (Start -> Run -> cmd) و دستورات `javac -version` و `java -version` را وارد کنید. اگر مراحل نصب را به درستی طی کرده باشید، باید پاسخی مانند پاسخ زیر دریافت کنید.



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>javac -version
javac 1.6.0

C:\Documents and Settings\Administrator>java -version
java version "1.6.0"
Java(TM) SE Runtime Environment (build 1.6.0-b105)
Java HotSpot(TM) Client VM (build 1.6.0-b105, mixed mode, sharing)

C:\Documents and Settings\Administrator>
```

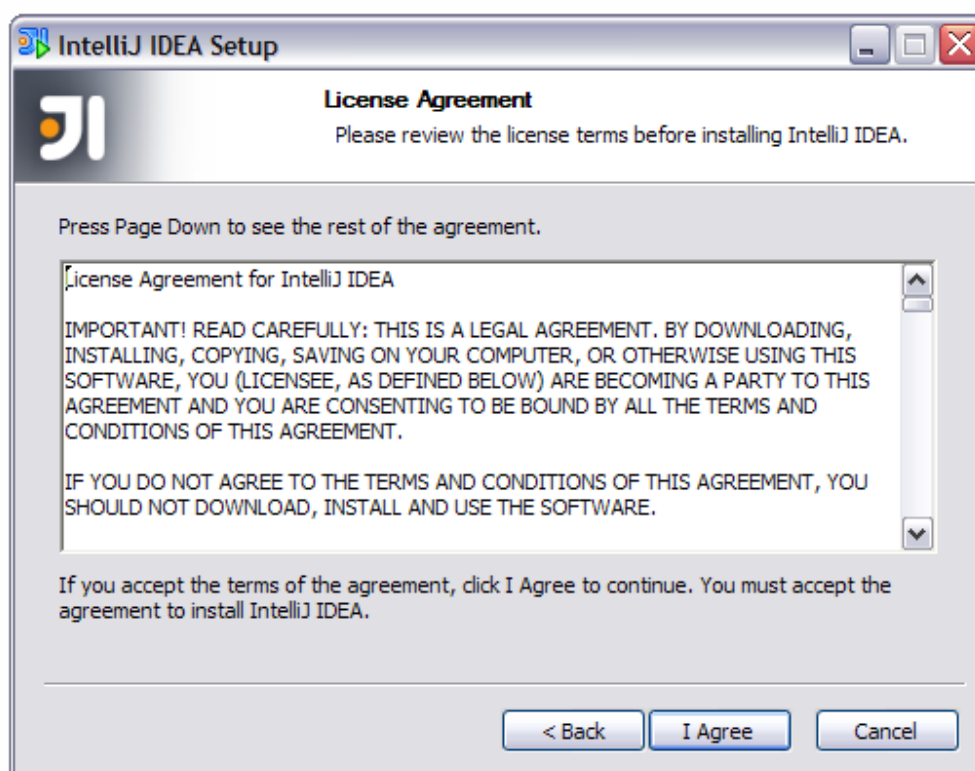
۲) نصب IntelliJ IDEA و ایجاد اولین پروژه

یکی از مهمترین ابزار برنامه نویسی به هر زبانی، محیطی است که در آن کد نویسی می‌کنید. این گونه نرم افزارها را غالباً IDE یا Integrated Development Environment می‌نامند. برای جاوا نیز IDE های مختلف با قابلیت‌های مختلف وجود دارد که از جمله آنها می‌توان به Eclipse، MyEclipse، NetBeans، JBuilder و IntelliJ IDEA اشاره نمود. در این قسمت مراحل نصب IntelliJ IDEA 8.0.1 توضیح داده می‌شود.

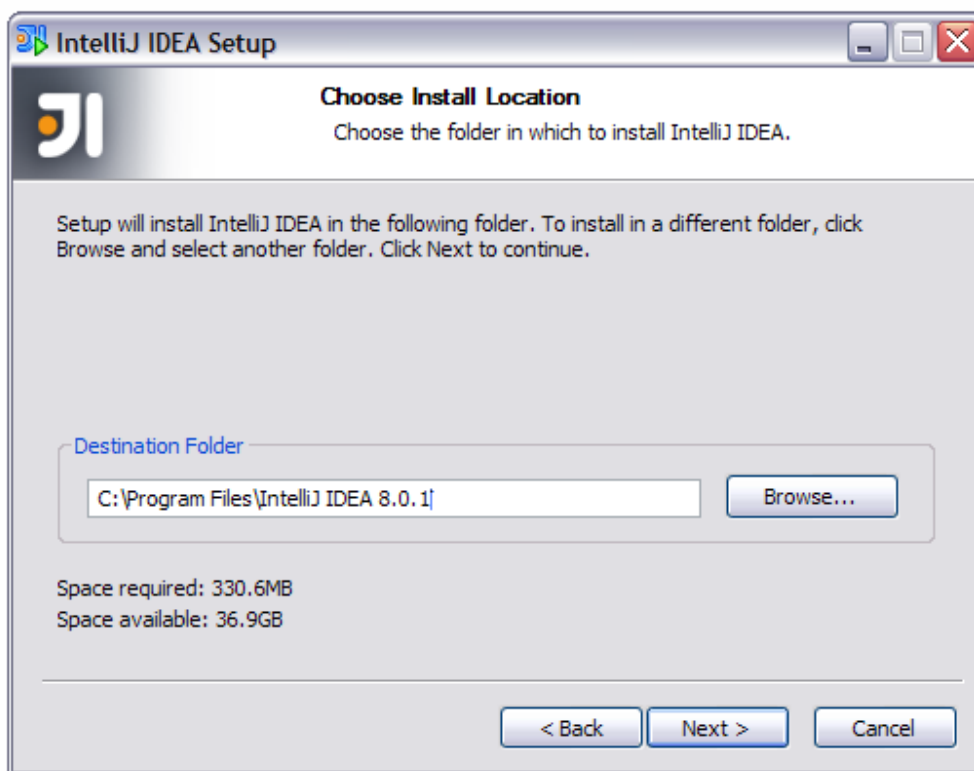
۱- بر روی فایل idea-8.0.1 دابل کلیک کنید تا پنجره زیر باز شود. بر روی دکمه Next > کلیک کنید.



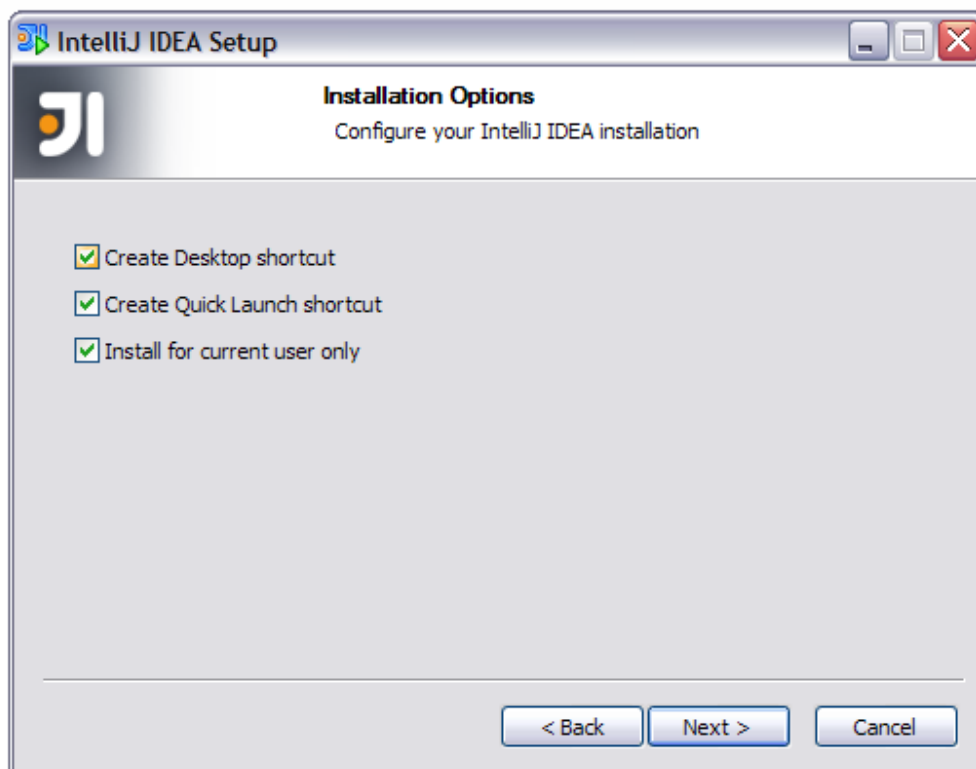
۲- بر روی I Agree کلیک کنید.



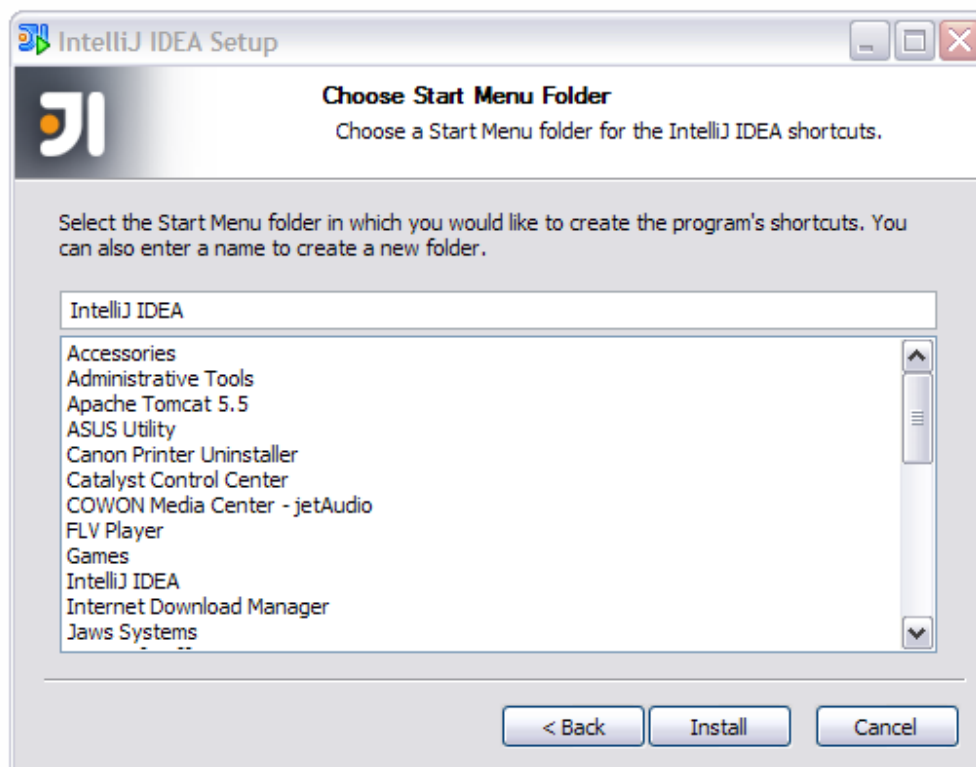
۳- در این پنجره می‌توانید مسیر نصب را تعیین کنید. من مسیر نصب را کمی تغییر داده‌ام ولی شما می‌توانید مسیر پیش‌فرض را پذیرفته و آن را تغییر ندهید.



۴- بر روی **Next >** کلیک کنید.



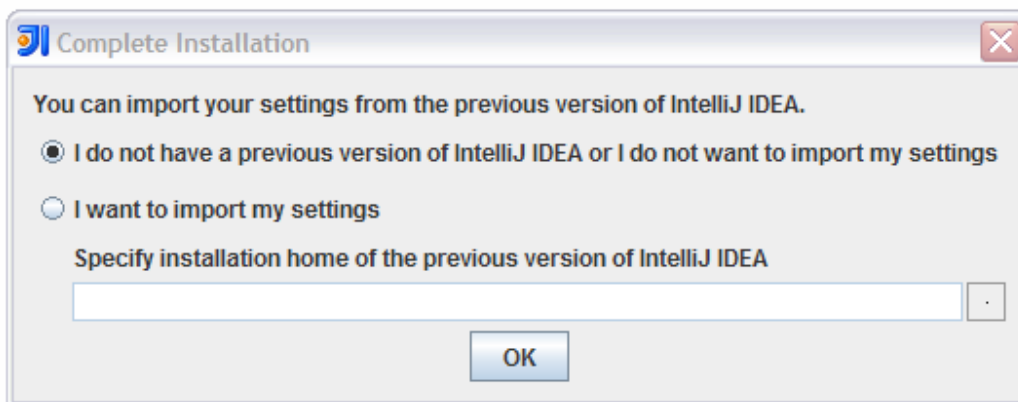
۵- بر روی Install کلیک کنید.



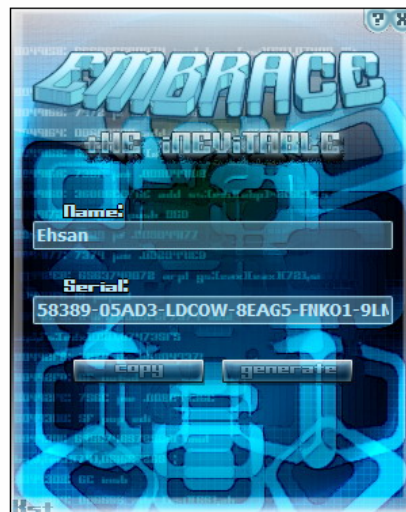
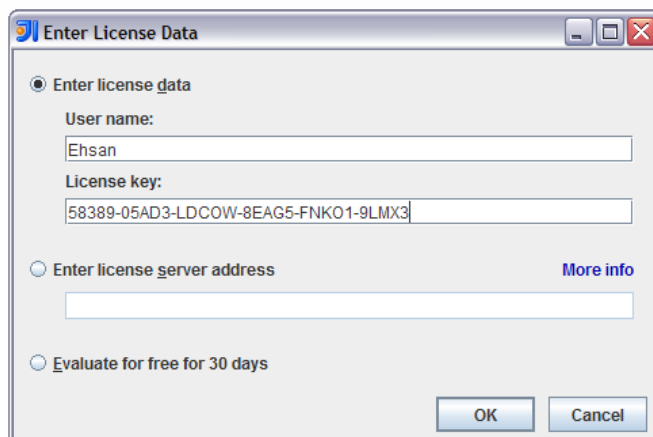
۶- در انتها نیز بر روی دکمه Finish کلیک کنید. اجازه دهید تا تیک مربوط به Run در IntelliJ IDEA خورده باشد.



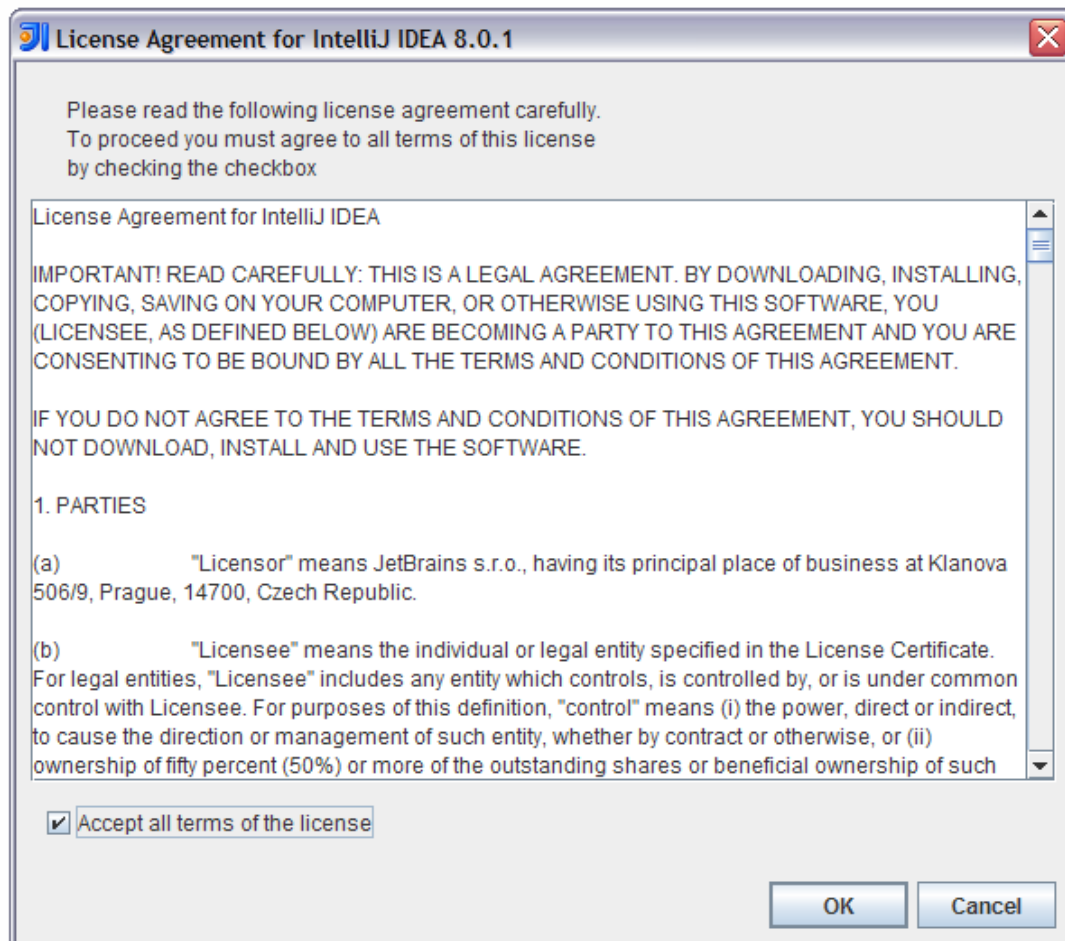
۷- اگر نسخه دیگری از IntelliJ را بر روی سیستم‌تان ندارید که بخواهید از تنظیمات آن نسخه برای نسخه جدید استفاده شود، گزینه اول را انتخاب نموده و بر روی OK کلیک کنید.



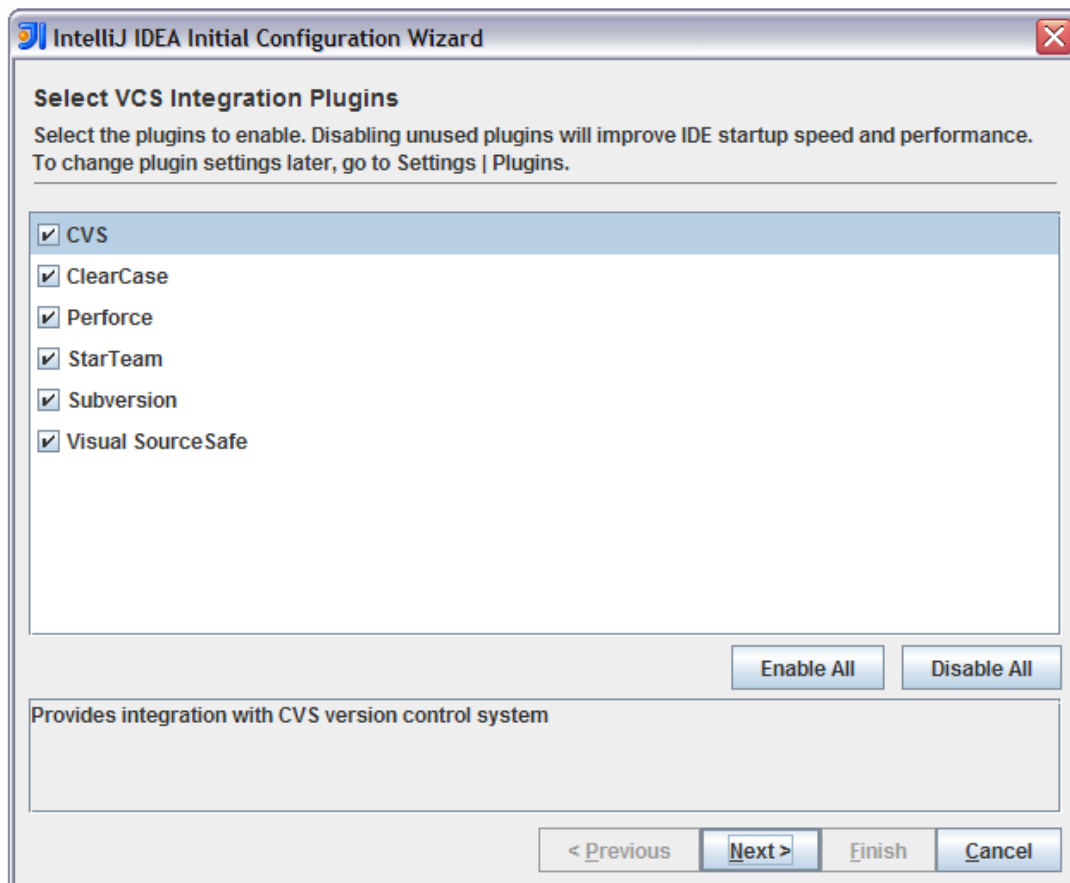
۸- در این قسمت می‌بایست اطلاعات مربوط به لیسانس نرم افزار وارد شود. به منظور بدست آوردن یک لیسانس، بر روی فایل `IntelliJ.IDEA.v8.0_keygen` دابل کلیک نموده و در قسمت `Name` یک نام کاربری وارد نموده، بر روی دکمه `generate` کلیک کنید تا بر اساس این نام کاربری یک شماره سریال ایجاد شود. بر روی دکمه `copy` کلیک کنید تا این شماره سریال وارد حافظه `Clipboard` سیستم شده و آن را در قسمت `License key` Paste کنید. (`Ctrl + V` را بزنید). در قسمت نام کاربری پنجره `Enter License data` نیز همان نام کاربری را وارد کنید. (برای نمونه در اینجا "Ehsan") در ادامه بر روی `OK` کلیک کنید و فایل `keygen` را هم می‌توانید ببندید.



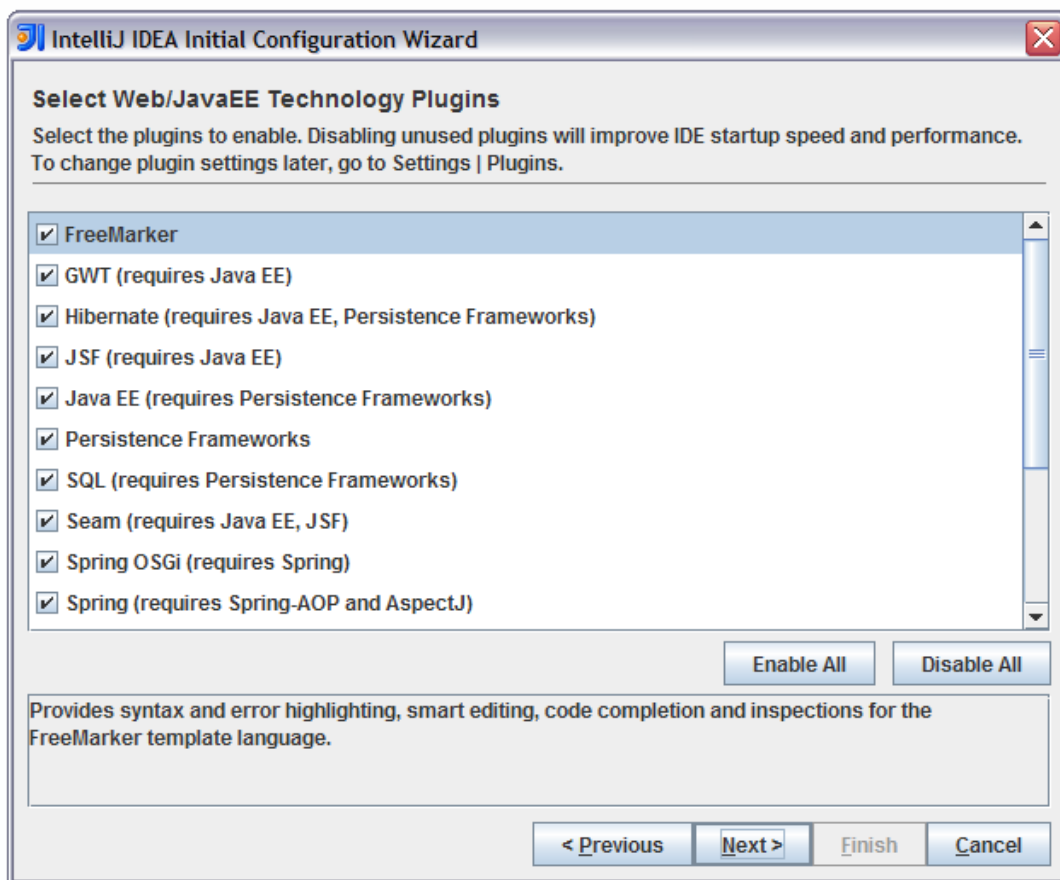
۹- در این پنجره، تیک مربوط به گزینه Accept all terms of license را علامت زده و OK را کلیک کنید.



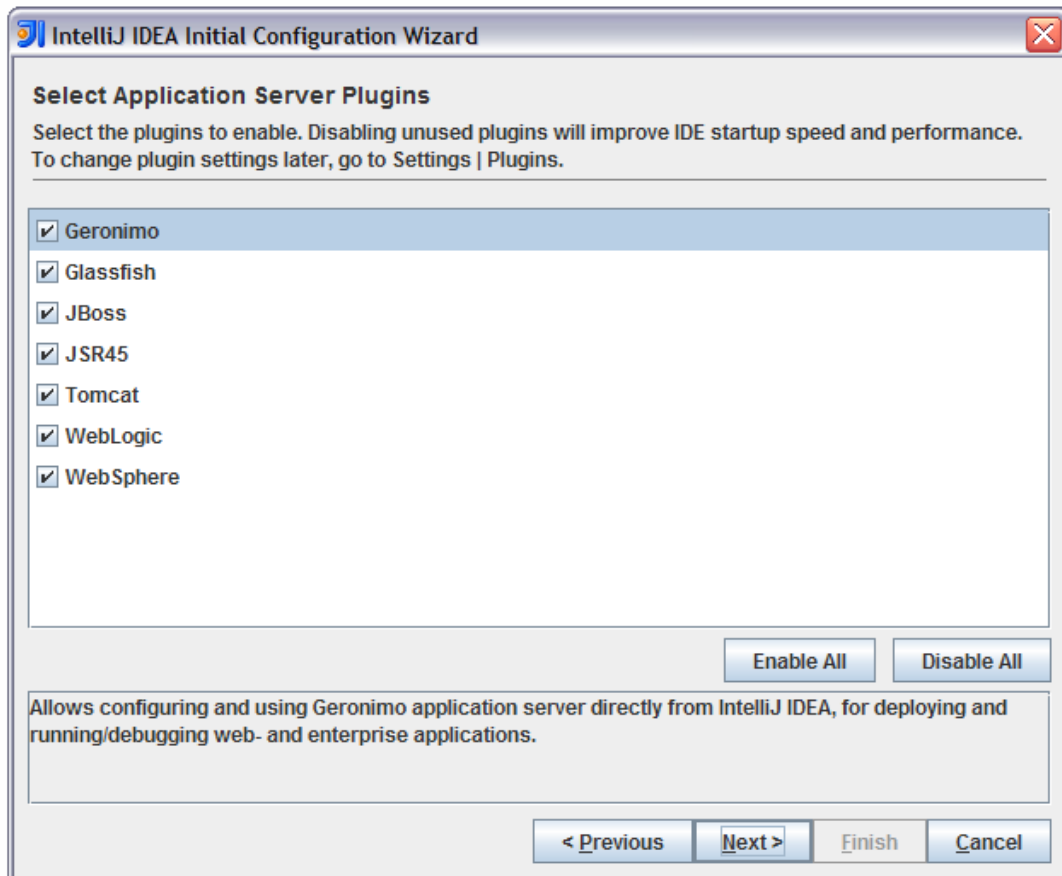
۱۰- در این پنجره نیز اجازه دهید تا پلاگینهای پیشفرض فعال باشند و بر روی دکمه Next > کلیک کنید.



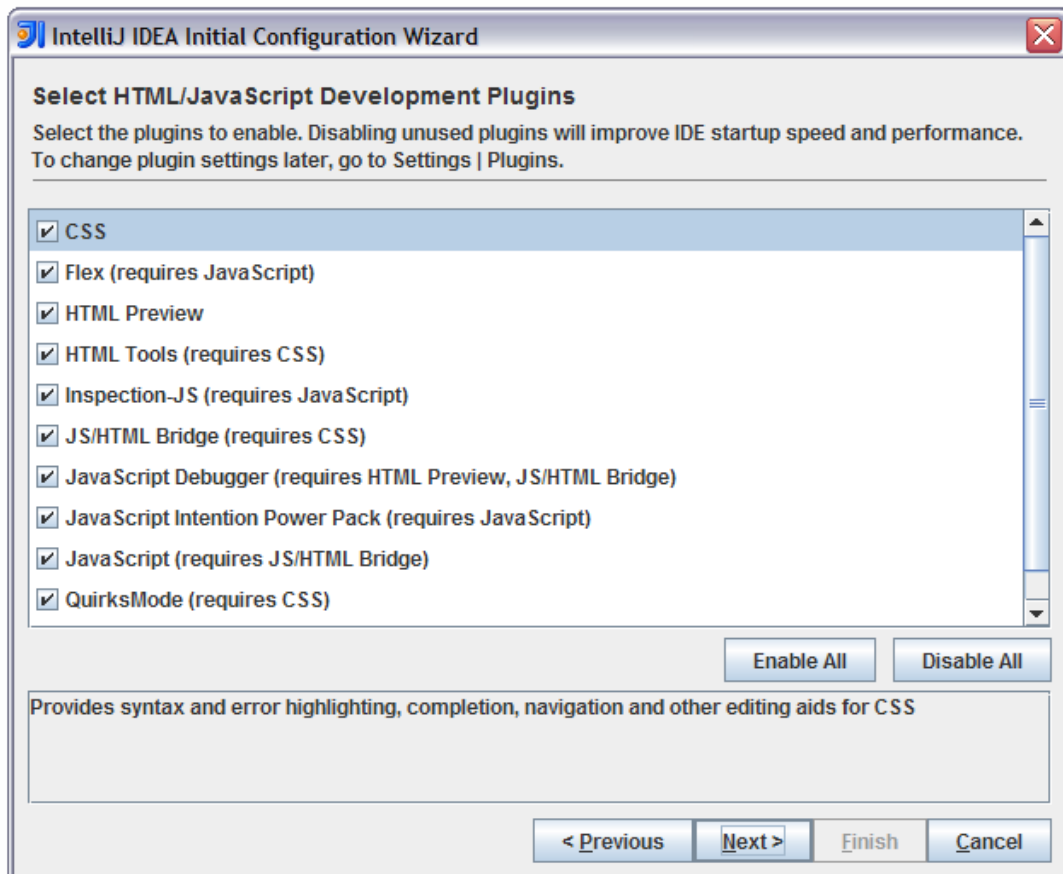
۱۱- در این پنجره نیز اجازه دهید تا پلاگینهای پیش فرض فعال باشند. هر کدام از این پلاگینها مربوط به یکی از تکنولوژیهای روز جاوا می باشند که با نصب شدن آنها، می توانید از امکانات و کلاسهای آنها، در برنامه خود استفاده کنید. (البته بیشتر این تکنولوژیها مربوط به Web Application می باشند مانند JSF، Struts، Struts2 و...) بر روی Next > کلیک کنید.



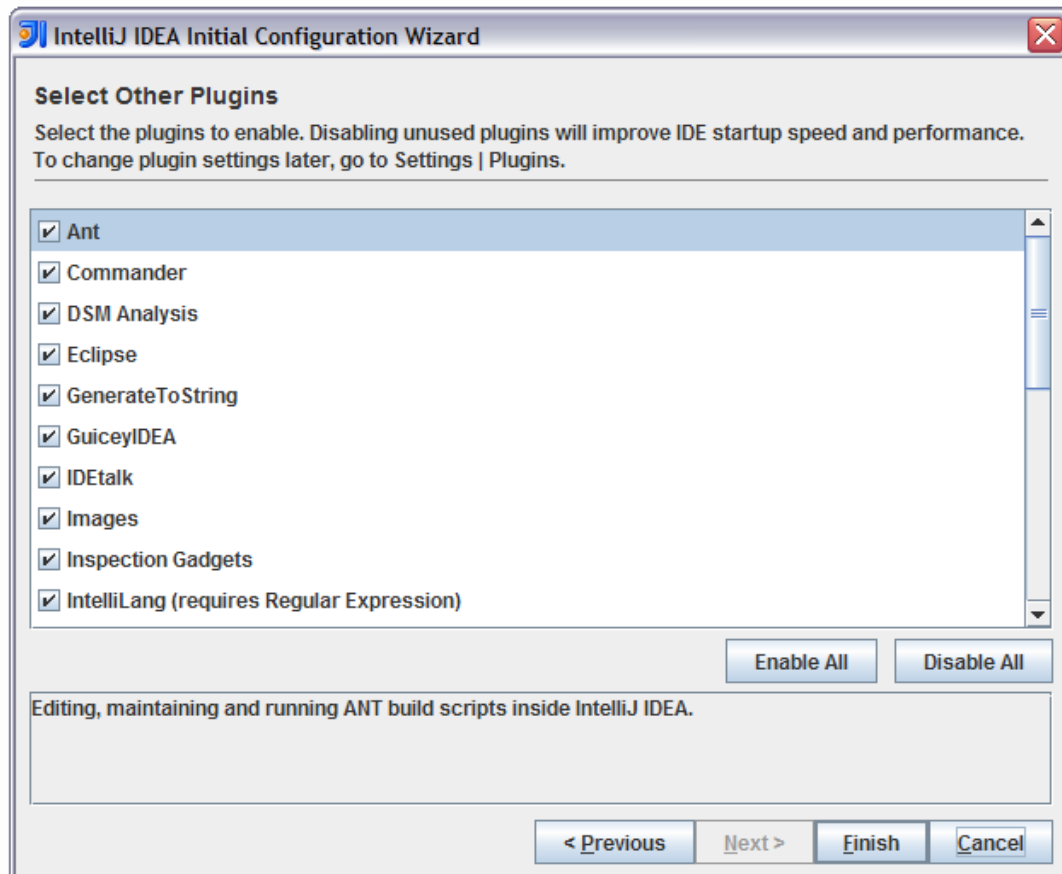
۱۲- در این پنجره نیز می‌توانید پلاگین‌های مربوط به سرورهای مختلف را نصب کنید. کسانی که تجربه ایجاد برنامه‌های تحت وب را دارند، به خوبی با سرورهای مختلف آشنا هستند) مقادیر پیش‌فرض را پذیرفته و بر روی **Next >** کلیک کنید.



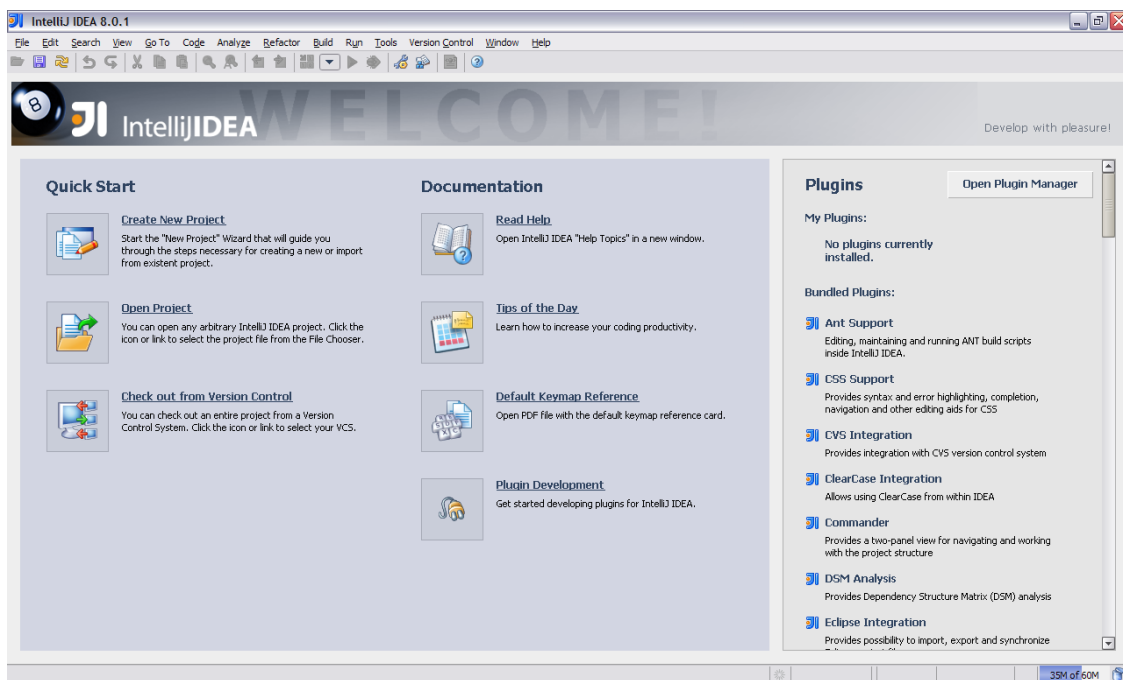
۱۳- در این پنجره نیز لیستی از پلاگینهای مختلف در رابطه با Web Development ارائه شده است. باز هم مقادیر پیش فرض را پذیرفته و بر روی **Next >** کلیک کنید.



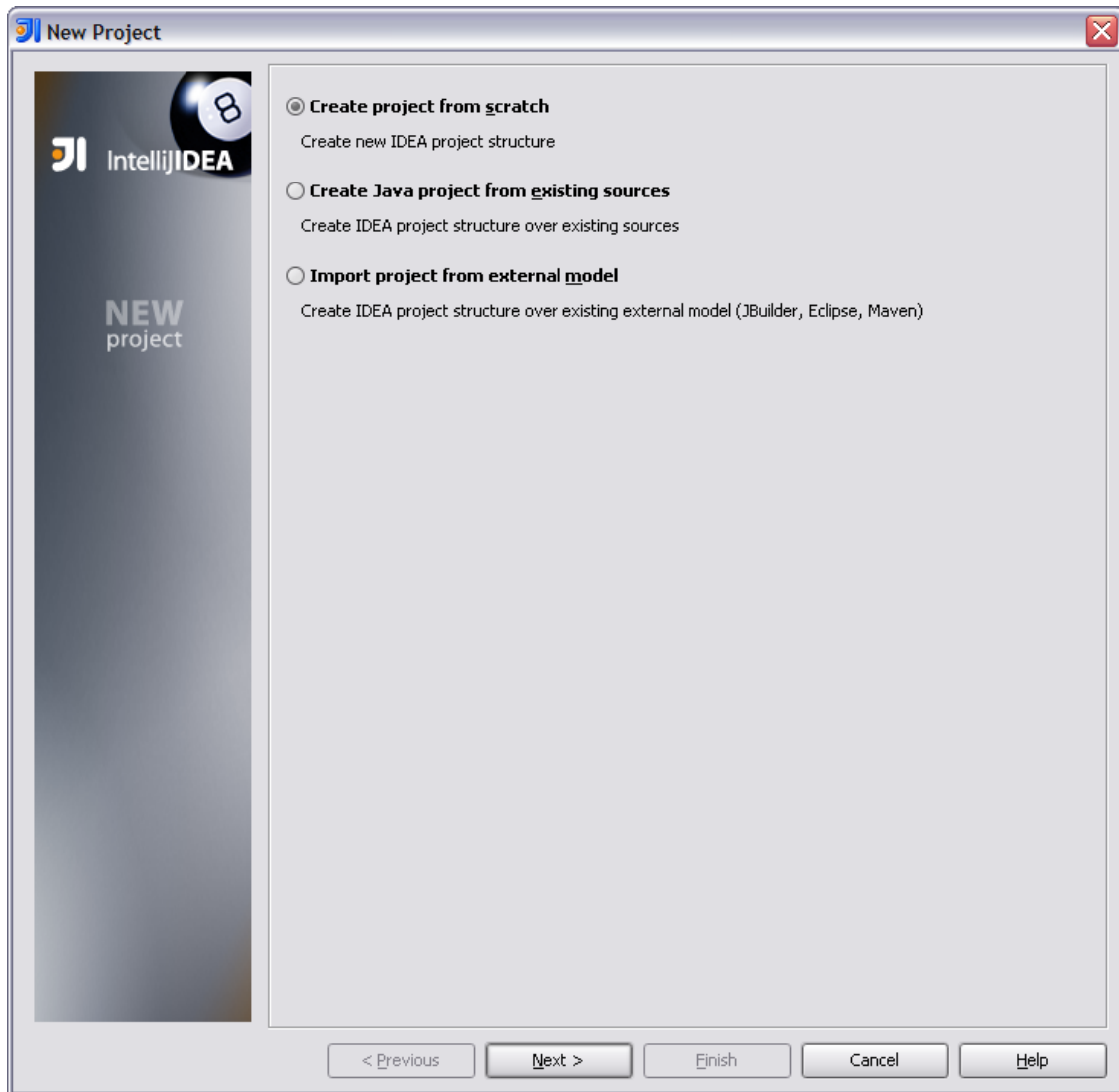
۱۴- در این پنجره نیز لیستی از پلاگینهای متفرقه ارائه شده است. مقادیر پیشفرض را پذیرفته و برروی Finish کلیک کنید.



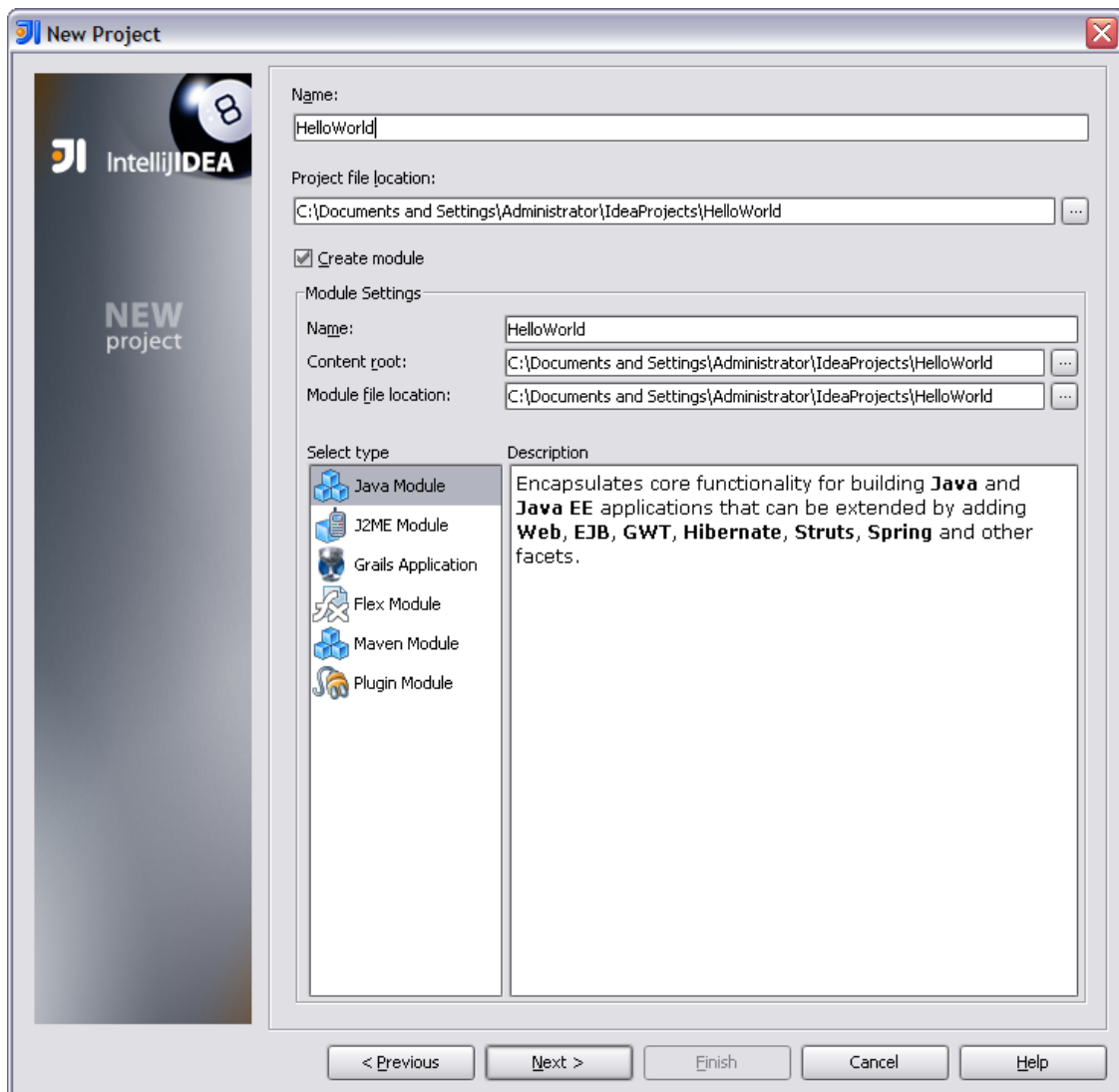
۱۵- در اینجا پنجره اصلی IntelliJ را مشاهده خواهید نمود. برای ایجاد پروژه جدید گزینه Create New Project و یا از منوی File گزینه New Project... را انتخاب کنید. برای باز نمودن پروژه‌ای که قبلاً ایجاد نموده‌اید، گزینه Open Project و یا از منوی File گزینه Open Project... را انتخاب کنید.



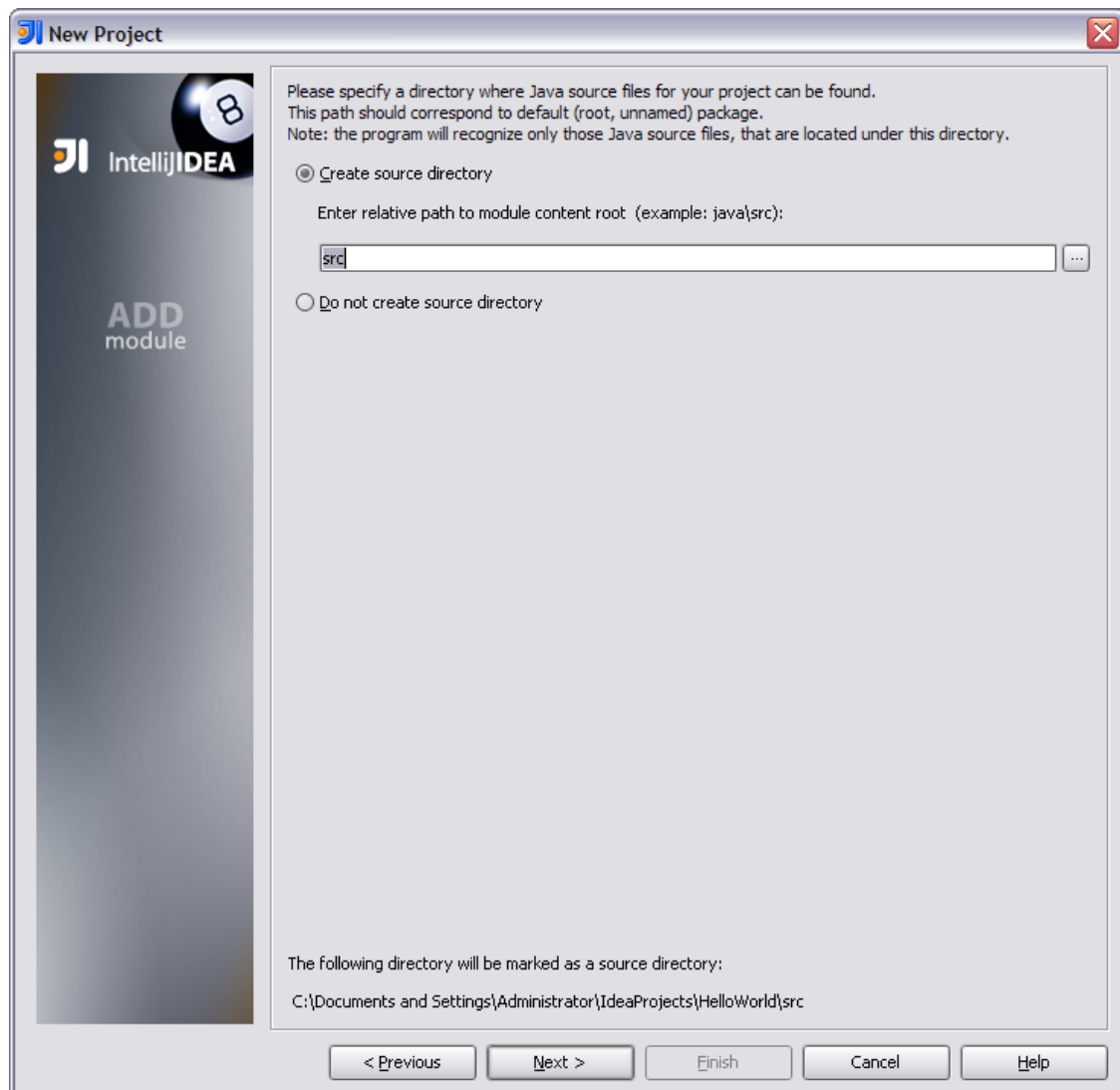
۱۶- به منظور ایجاد پروژه‌ای جدید و از صفر، گزینه اول را انتخاب نموده و > Next را انتخاب کنید.




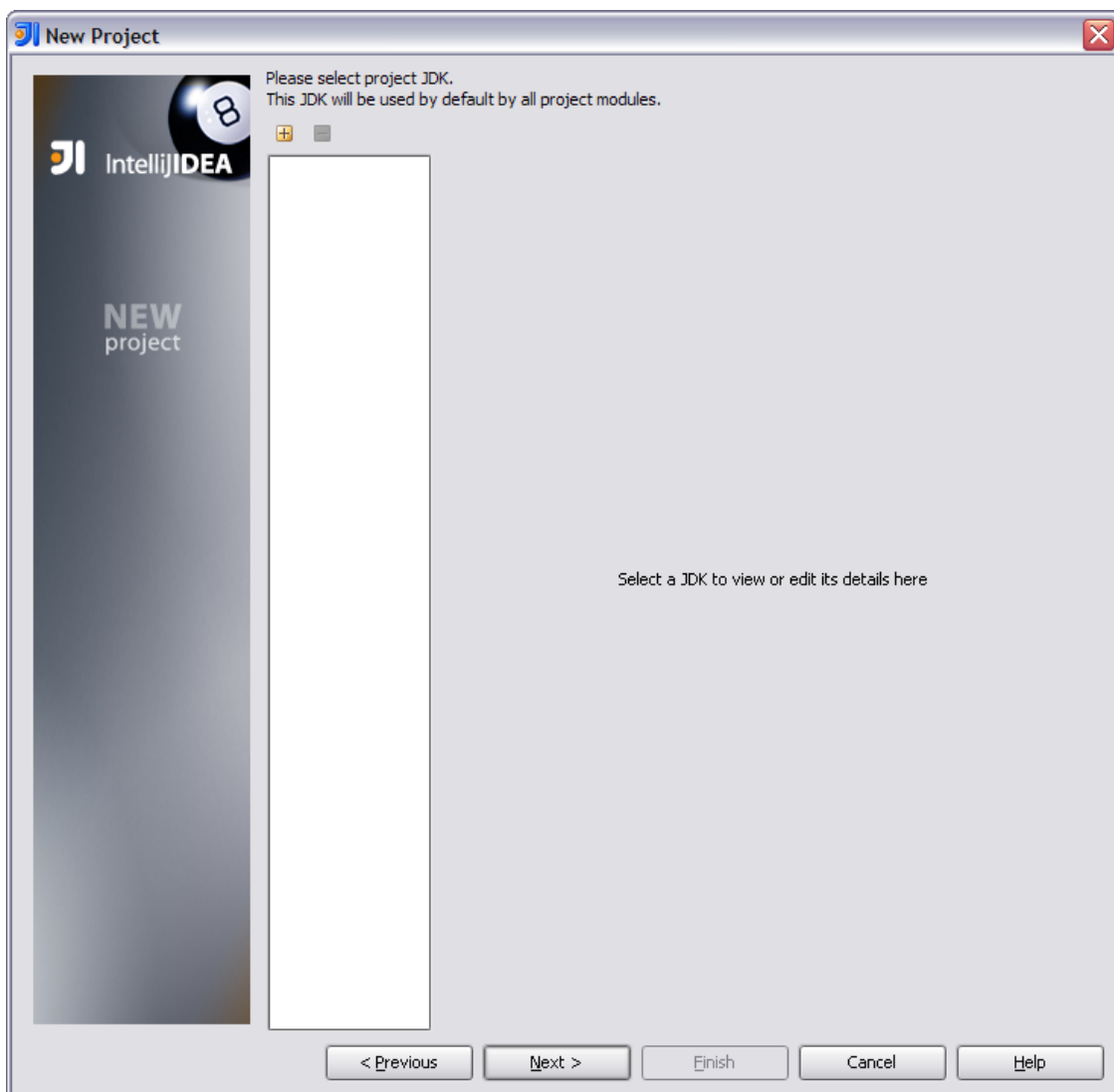
۱۷- در این پنجره می‌بایست برای پروژه خود نامی انتخاب نموده، مسیر ذخیره فایل‌های آن را تعیین کنید و نوع پروژه را انتخاب کنید. برای نمونه اینجا قصد داریم تا یک پروژه کوچک تحت Desktop ایجاد کنیم. بدین منظور در پانل Select type گزینه Java Module را انتخاب می‌کنیم. بر روی > Next کلیک کنید.



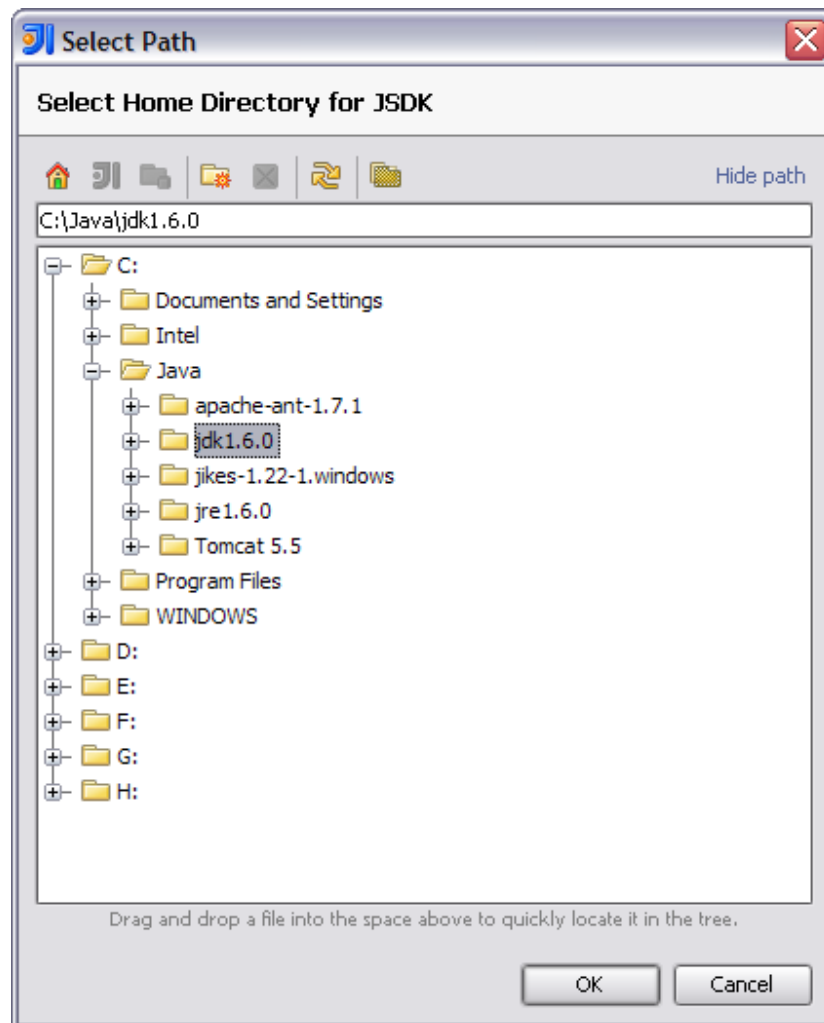
۱۸- در این پنجره نیز می‌بایست دایرکتوری اصلی پروژه که پکیجها و کلاسهای که ایجاد خواهیم کرد، در آن قرار خواهند گرفت را تعیین کنیم. اجازه دهید تا نام این دایرکتوری همان مقدار پیش فرض یعنی `src` باشد و بر روی `Next >` کلیک کنید.



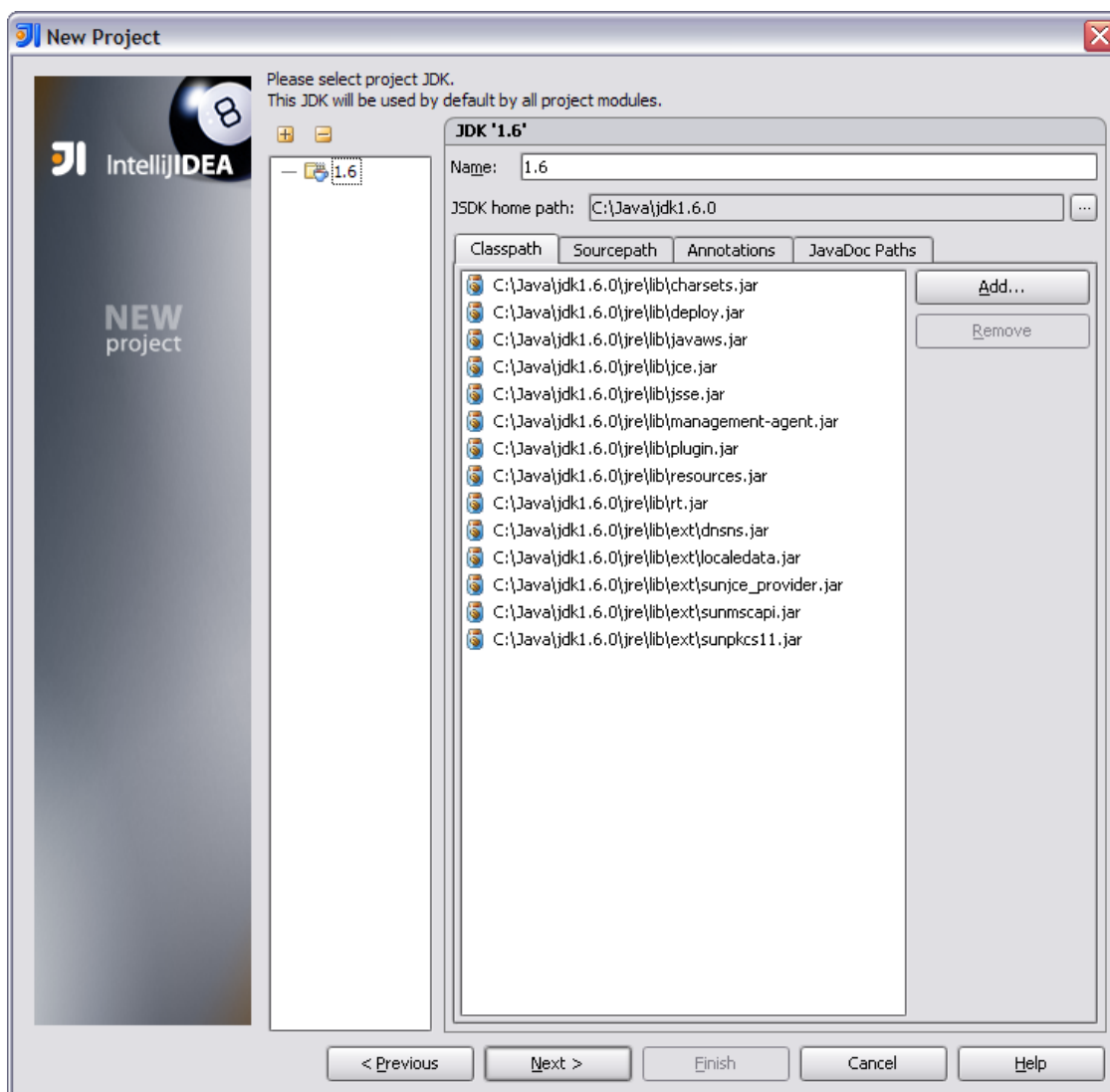
۱۹- در این پنجره لازم است تا مسیر نصب JDK را برای IntelliJ تعیین کنید. بدین منظور بر روی دکمه  کلیک نموده، از میان گزینه‌های موجود JSDK را انتخاب و مسیر نصب JDK را تعیین کنید.



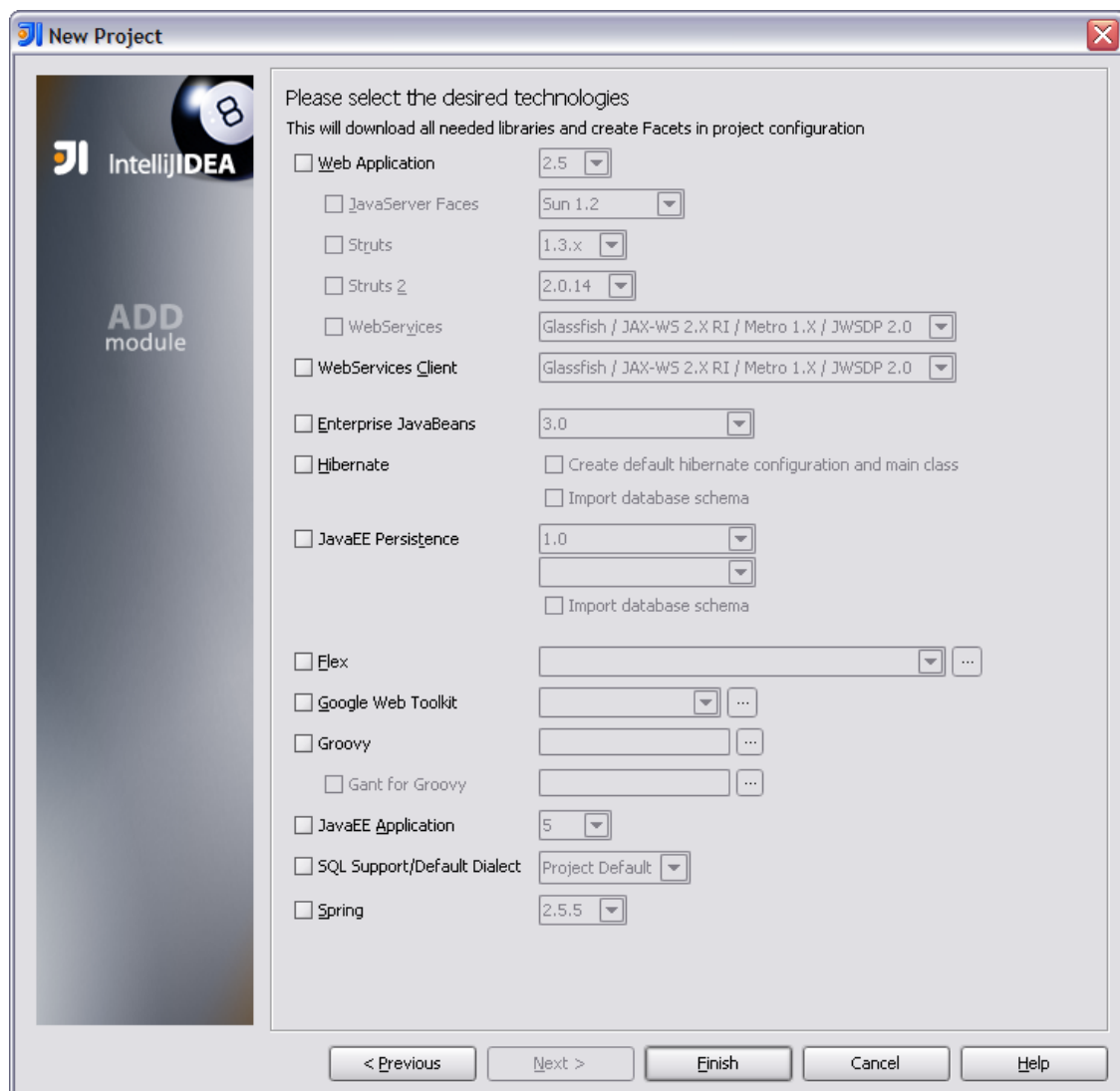
در پنجره زیر پس از تعیین مسیر OK را کلیک کنید.



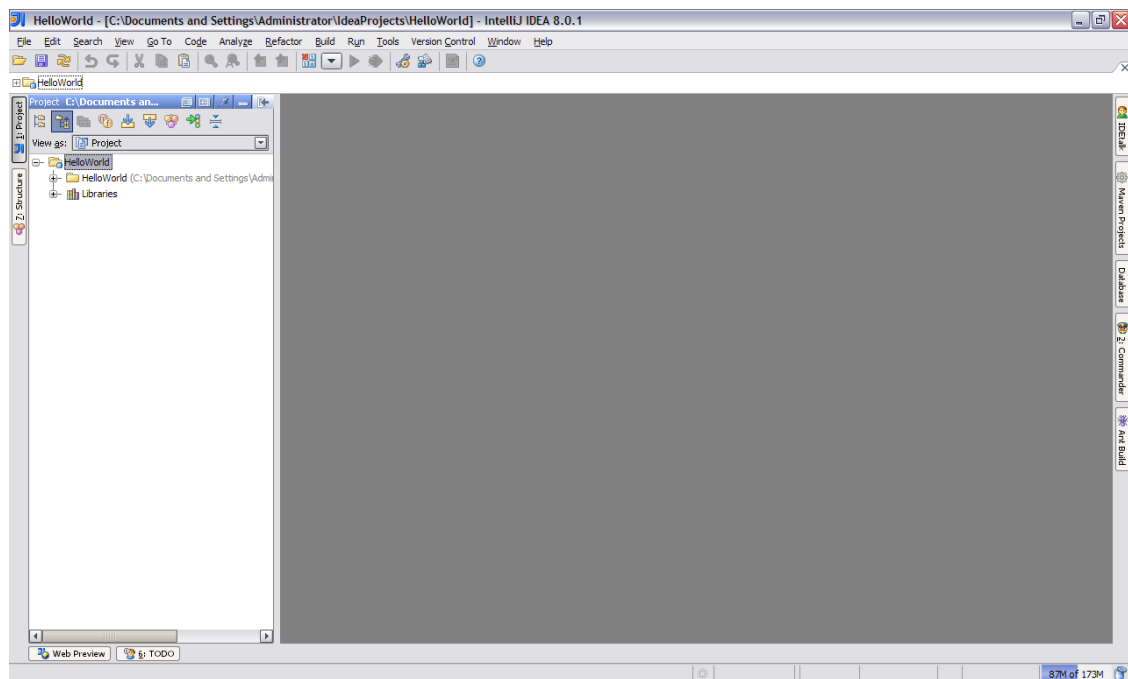
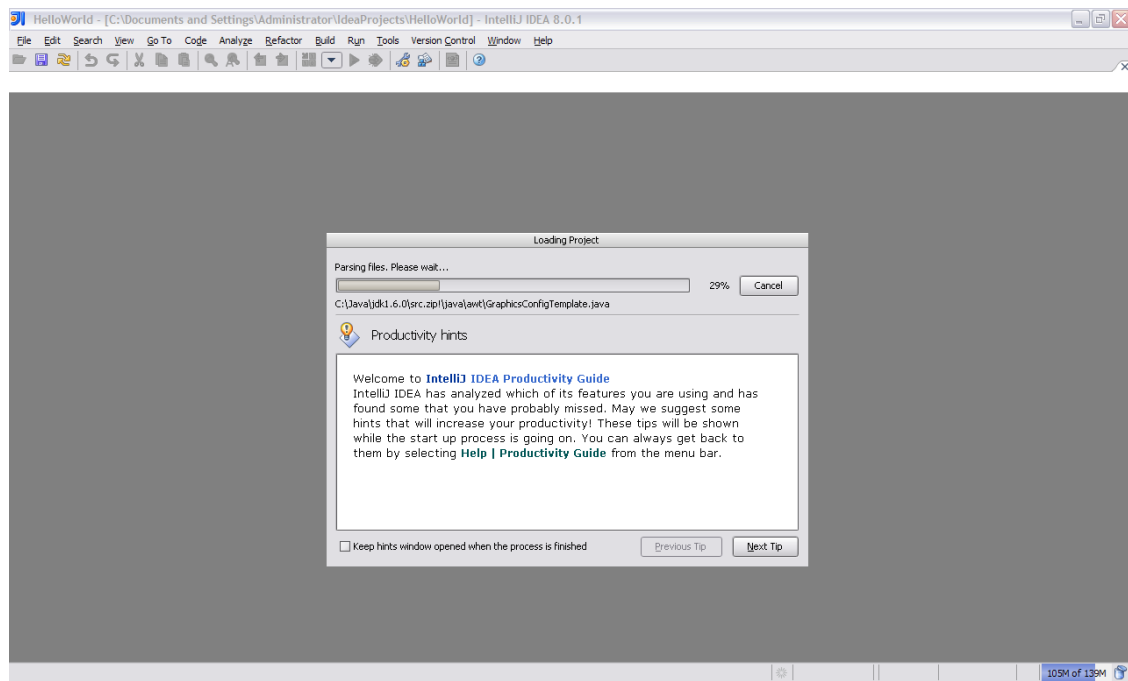
با اینکار، فایل‌های jar لازم که به همراه JDK ارائه شده‌اند، لیست می‌شوند. بر روی Next > کلیک کنید.



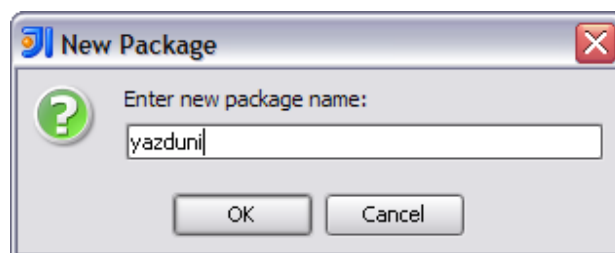
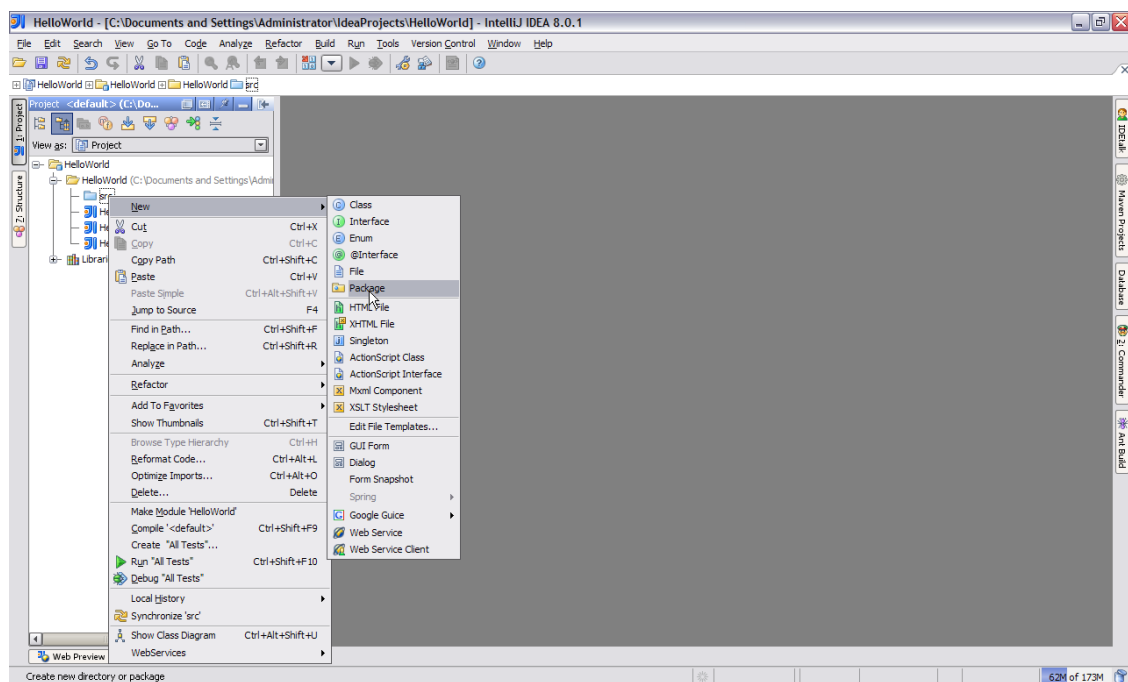
۲۰- در این پنجره می‌توانید از میان تکنولوژی‌های مختلفی که پلاگین‌های آنها را در زمان نصب IntelliJ نصب کردید، انتخاب کنید تا فایل‌های jar آنها به پروژه‌تان اضافه شده و بتوانید از آنها در پروژه خود استفاده کنید. همانطور که قبلاً ذکر شد، بیشتر این تکنولوژی‌ها مربوط به Web Application Development بوده و فعلاً با آنها کاری نداریم. بر روی Finish کلیک کنید.



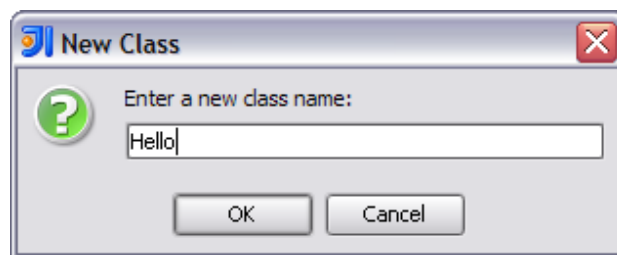
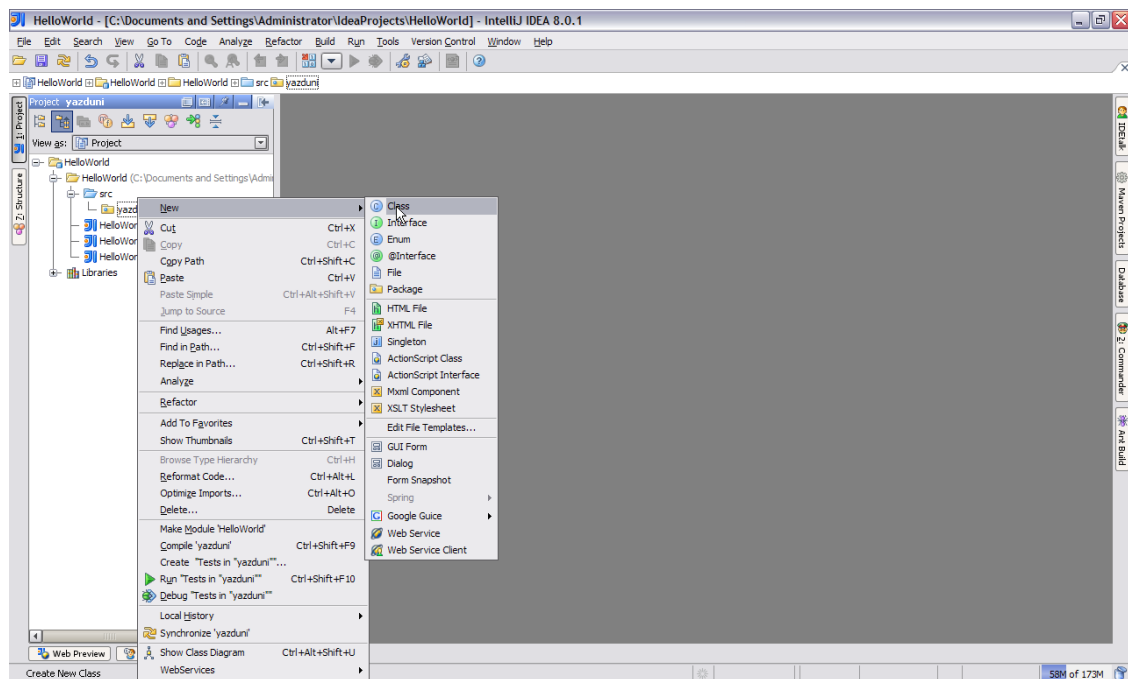
۲۱- در ادامه کمی طول می کشد تا فایل های پروژه تان بارگذاری شود. بعد از باز شدن پروژه، صفحه ای مانند تصویر دوم را مشاهده خواهید نمود.



۲۲- لازم است تا پروژه‌تان (برای برنامه های تحت دسکتاپ یا Desktop Applications) حداقل دارای کلاسی با تابع main باشد. برای نمونه در ابتدا پکیجی با نام yazduni ایجاد می‌کنیم. بدین منظور در پانل سمت چپ، بر روی پوشه src کلیک راست نموده و از گزینه New گزینه Package را انتخاب کنید و نام yazduni را در کادر محاوره‌ای وارد کنید.




۲۳- به منظور ایجاد کلاس جدید، بر روی پکیج ایجاد شده کلیک راست نموده و از گزینه New گزینه Class را انتخاب کنید. در کادر محاوره‌ای باز شده، نامی برای کلاس خود تعیین کنید. برای نمونه، در اینجا Hello

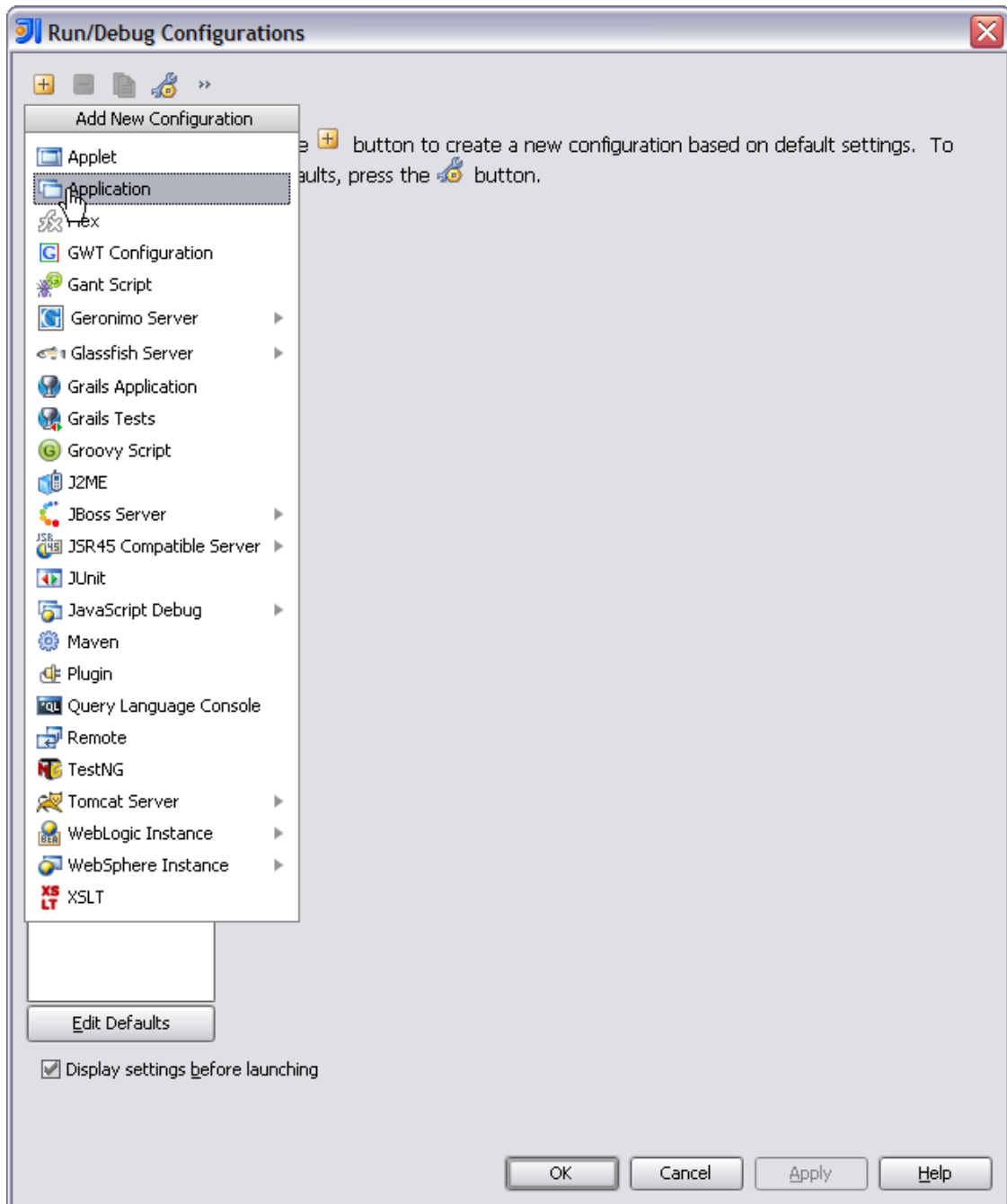


۲۴- بعد از ایجاد کلاس جدید، می‌توانیم method ها و property های مختلفی برای آن بنویسیم. در اینجا، متد main را با امضای `public static void main(String[])` ایجاد می‌کنیم. این متد، متدی است که ماشین مجازی جاوا آن را به عنوان نقطه ورود به برنامه‌تان دانسته و اجرای پروژه‌تان را از آن شروع می‌کند. لازم است تا حداقل یکی از کلاسهای پروژه‌تان حاوی متد main با همین امضاء باشد.

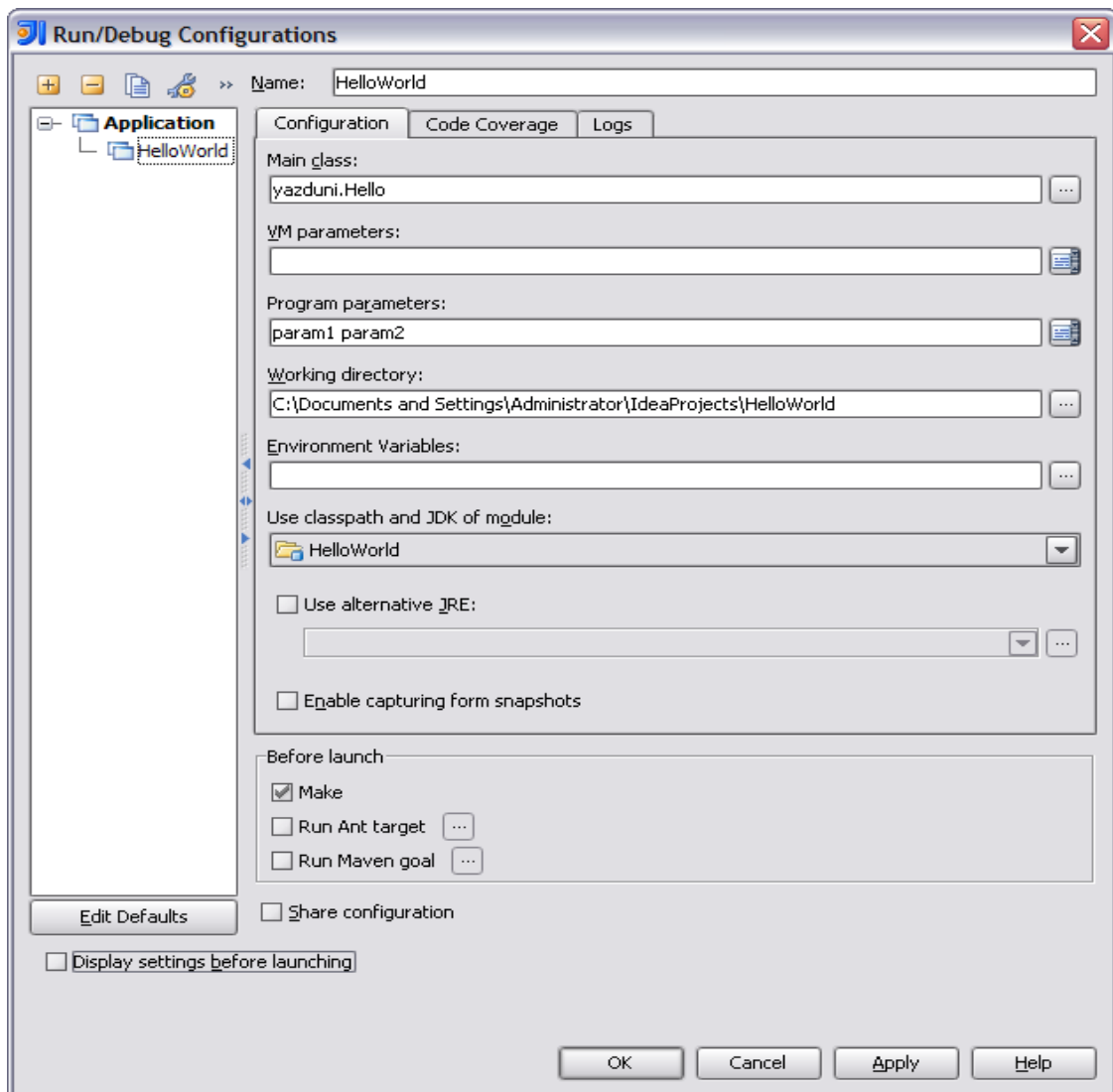
```
package yazduni;


/**
 * Created by IntelliJ IDEA.
 * User: Administrator
 * Date: Feb 11, 2010
 * Time: 5:21:20 PM
 * To change this template use File | Settings | File Templates.
 */
public class Hello {
    public static void main(String[] args){
        System.out.println("Hi guys, Welcome to Java World!");
    }
}
```

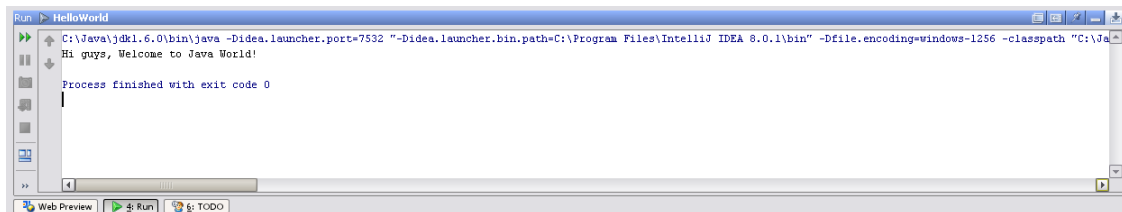
۲۵- برای اجرای برنامه در دفعه اول لازم است تا یک سری تنظیمات انجام شود. از منوی Run گزینه Edit Configurations را انتخاب کنید. در پنجره زیر دکمه  را کلیک نموده و از میان گزینه‌های موجود، Application را انتخاب کنید.



۲۶- در پنجره زیر، در بخش Name نامی را تعیین کنید (بهتر است همان نام پروژه باشد)، در بخش Main class کلاس اصلی پروژه که حاوی متد main است را با کلیک بر روی دکمه [...] انتخاب کنید. در بخش Program parameters هم می‌توانید آرگومانهای لازم را در قالب آرایه‌ای از رشته‌ها به متد main ارسال کنید. (اگر چندین آرگومان دارید، آنها را با فاصله از هم جدا نموده و در متد main نیز از طریق پارامتر args از آنها استفاده کنید.) اگر می‌خواهید در هر بار اجرای برنامه این پنجره را نبینید، تیک Display setting before launching را بردارید. بر روی OK کلیک کنید.



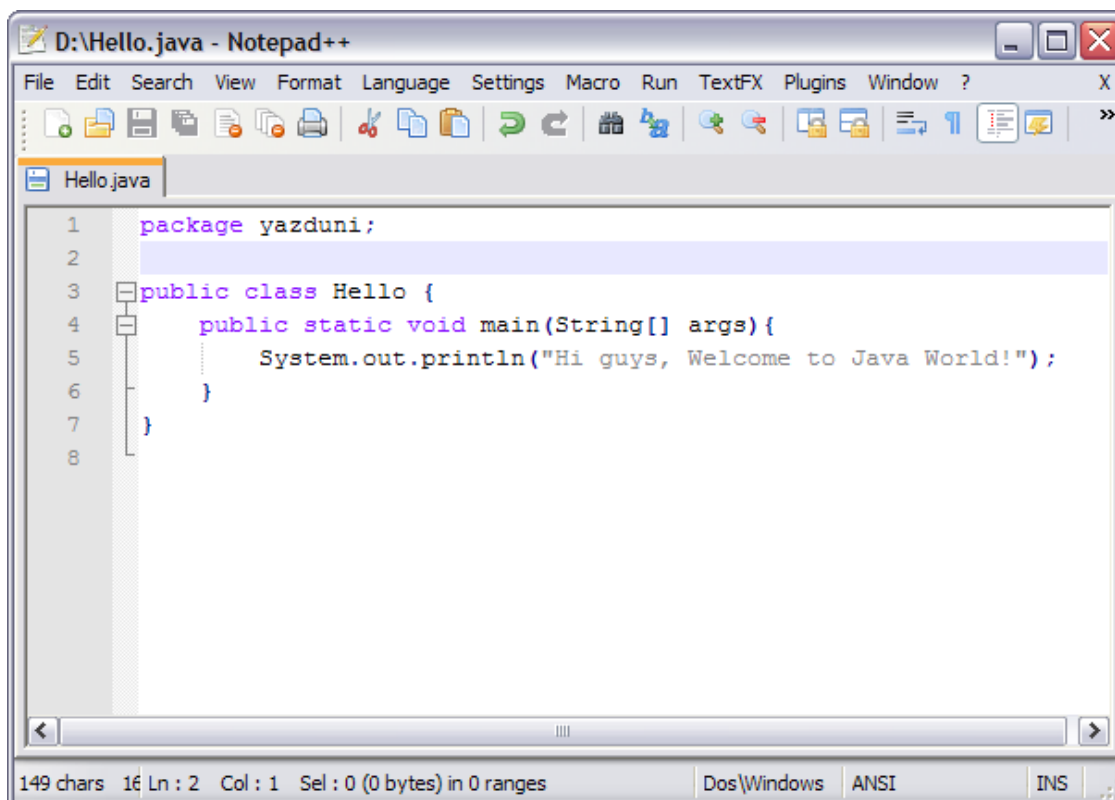
۲۷- حال شما می‌توانید برنامه خود را اجرا کنید. بدین منظور می‌توانید دکمه  را کلیک نموده و یا دکمه Shift + F10 را فشار دهید. برنامه اجرا شده و خروجی زیر را در Console مشاهده خواهید نمود.



تبریک عرض می‌کنم. شما توانستید برای اولین بار یک برنامه جاوا را ایجاد نموده و آن را اجرا کنید. این نقطه شروع به دنیای Java بود و هنوز راه درازی در پیش دارید. به همین دلیل توصیه می‌کنم با شروع از یک کتاب مناسب، با پایه های زبان جاوا آشنا شده و با کدنویسی های مکرر و ایجاد پروژه های مختلف، خود را در زمره یکی از برنامه نویسان حرفه ای این زبان قرار دهید.

۳) کامپایل و اجرای برنامه ها از خط Command

در این بخش قصد داریم تا نحوه ایجاد و اجرای برنامه ها از خط command را توضیح دهیم. فرض کنید، کلاسی به نام Hello.java را در مسیر D:\ ایجاد نموده ایم.

A screenshot of the Notepad++ application window. The title bar reads "D:\Hello.java - Notepad++". The menu bar includes File, Edit, Search, View, Format, Language, Settings, Macro, Run, TextFX, Plugins, Window, and ?. The toolbar contains various icons for file operations and editing. The text area shows the following Java code:

```
1 package yazduni;  
2  
3 public class Hello {  
4     public static void main(String[] args){  
5         System.out.println("Hi guys, Welcome to Java World!");  
6     }  
7 }  
8
```

The status bar at the bottom indicates "149 chars 16 Ln : 2 Col : 1 Sel : 0 (0 bytes) in 0 ranges" and shows encoding options "Dos\Windows", "ANSI", and "INS".

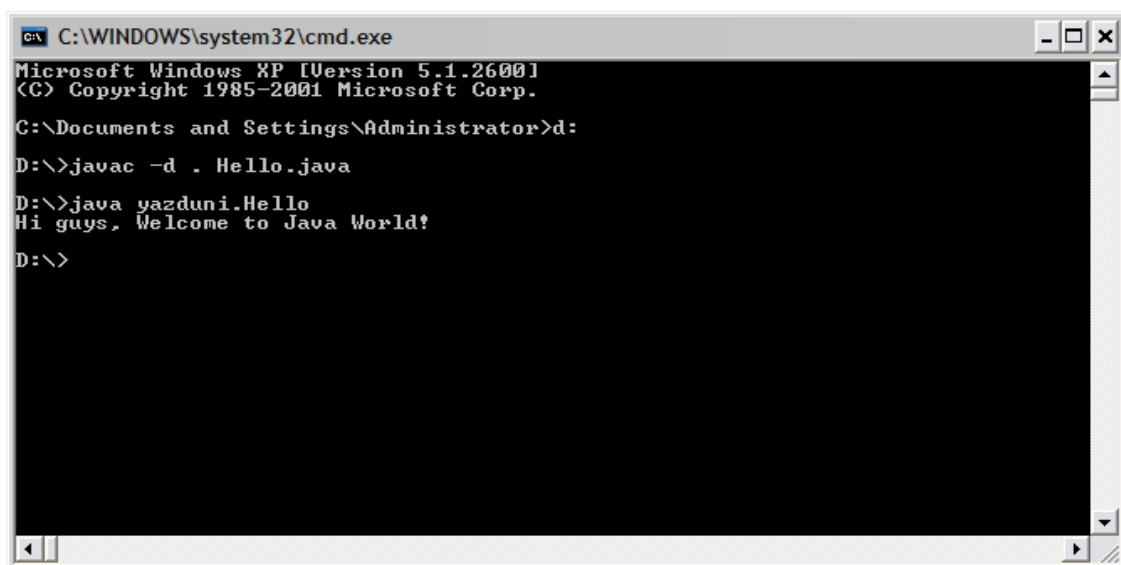
در ابتدا از طریق پنجره command prompt به درایو D:\ و محل استقرار کلاس خود حرکت می کنیم. در آنجا دستور زیر را وارد می کنیم تا علاوه بر کامپایل شدن کلاس به Bytecode و ایجاد فایل class. متناظر، دایرکتوریهای متناظر با پکیجی که کلاس در آن قرار دارد (در این مثال "yazduni" ایجاد شود:

`javac -d . [نام کلاس].java`

با اجرای دستور فوق به ازای پکیجهایی که کلاس در آن قرار دارد دایرکتوریهای لازم ایجاد می شود. برای نمونه در این مثال کلاس Hello در پکیج yazduni قرار دارد.

بنابراین یک دایرکتوری به نام yazduni ایجاد می‌شود که فایل Hello.class در آن قرار خواهد داشت.

در ادامه به منظور اجرای برنامه کافایت تا دستور زیر را اجرا کنید. دقت شود که برای اجرای برنامه لازم نیست که به داخل پکیج cd کنید. برای نمونه در اینجا لازم نیست تا به داخل دایرکتوری yazduni حرکت کنید. فقط کافایت تا هم سطح آن ایستاده و دستور java را همراه با مسیر کامل کلاس (یعنی نام پکیج.نام کلاس) اجرا کنید.



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>d:
D:\>javac -d . Hello.java
D:\>java yazduni.Hello
Hi guys, Welcome to Java World!
D:\>
```


به منظور رفع هرگونه ابهام در این زمینه مثال دیگری می‌زنم. فرض کنید که پروژه خود را در IntelliJ ایجاد و کامپایل نموده‌اید و قصد دارید برای اجرای آن از خط دستور استفاده کنید. از همان مثالی که در IntelliJ ایجاد نموده‌اید، استفاده می‌کنم. از خط command استفاده نموده و به دایرکتوری متناظر با پکیج حرکت کنید. پروژه‌های IntelliJ بصورت پیش‌فرض در مسیر زیر قرار می‌گیرند:


C:\Documents and Settings\Administrator\IdeaProjects

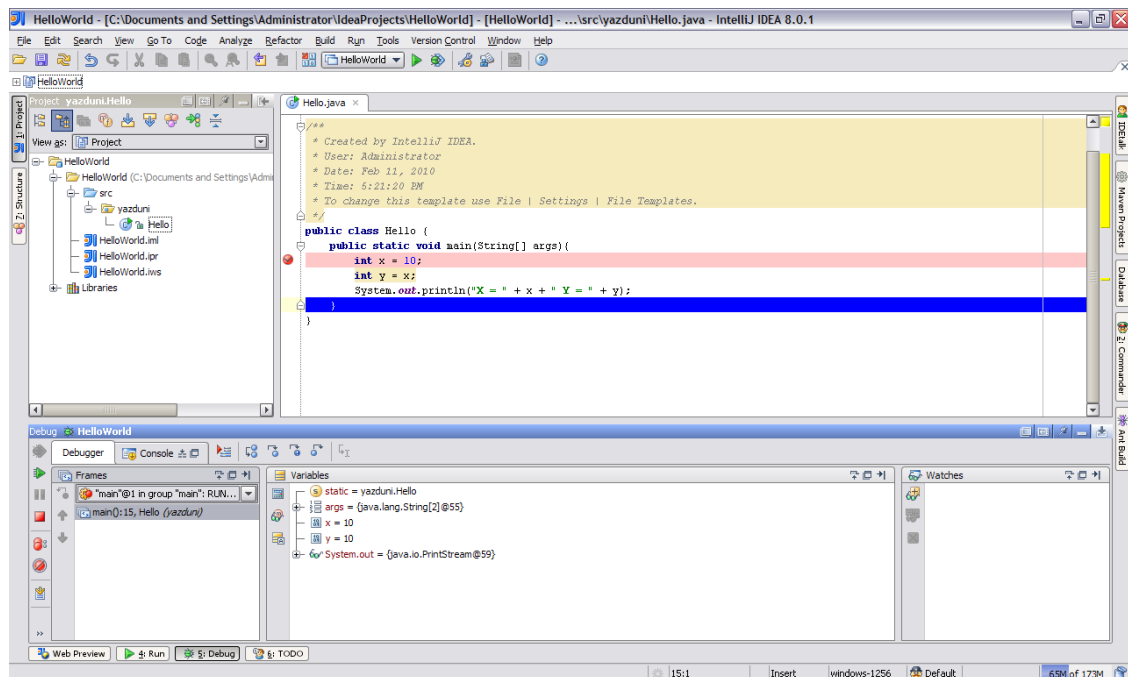
رشته دستورات لازم برای حرکت به دایرکتوری HelloWorld را مانند تصویر زیر وارد کنید. دقت شود که دایرکتوری‌های out و production توسط IntelliJ ایجاد شده‌اند. در همان دایرکتوری HelloWorld -> production -> out توقف نموده و دستور java yazduni.Hello را وارد کنید.

```
Select C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

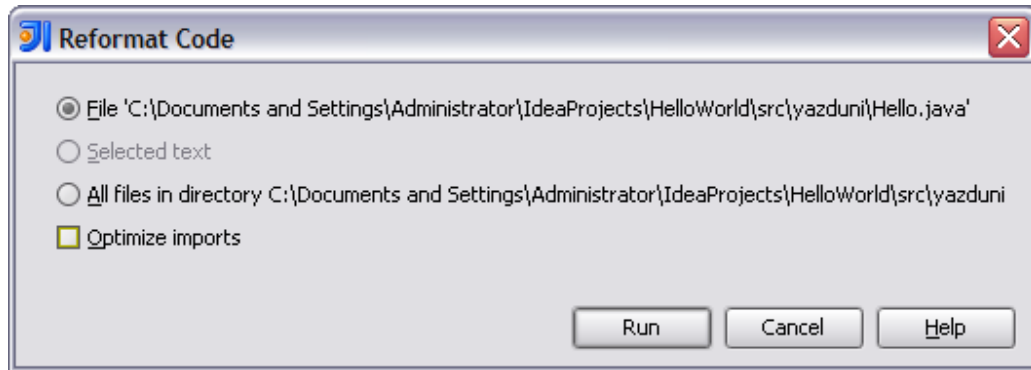
C:\Documents and Settings\Administrator>cd IdeaProjects
C:\Documents and Settings\Administrator\IdeaProjects>cd HelloWorld
C:\Documents and Settings\Administrator\IdeaProjects\HelloWorld>cd out
C:\Documents and Settings\Administrator\IdeaProjects\HelloWorld\out>cd production
C:\Documents and Settings\Administrator\IdeaProjects\HelloWorld\out\production>cd HelloWorld
C:\Documents and Settings\Administrator\IdeaProjects\HelloWorld\out\production\HelloWorld>java yazduni.Hello
Hi guys, Welcome to Java World!
C:\Documents and Settings\Administrator\IdeaProjects\HelloWorld\out\production\HelloWorld>
```

۴) نکات متفرقه

➤ برای Debug نمودن پروژه خود در IntelliJ و Trace نمودن بر روی کد، ابتدا بر روی خط شروع متد، یک breakpoint ایجاد کنید و سپس بر روی دکمه  کلیک کنید و یا Shift + F9 را فشار دهید. برای ایجاد و یا حذف breakpoint کافیست تا بر روی حاشیه Editor کلیک کنید. برای Trace نمودن نیز از دکمه‌های F7 یا F8 استفاده کنید. مقادیر متغیرها و یا object ها در پانل انتهای IntelliJ نمایش داده می‌شود.



- برای قالب دهی به source code خود و زیباتر نمودن آن از دکمه‌های **Ctrl + Alt + L** استفاده نموده و **Run** را کلیک کنید.



- برای **Comment** نمودن قطعه‌ای از کد، آن را **Highlight** نموده و **Ctrl + Shift + /** را فشار دهید.


- برای جستجو در فایل جاری از **Ctrl + F** و برای جستجو در کل پروژه از **Ctrl + Shift + F** استفاده کنید.

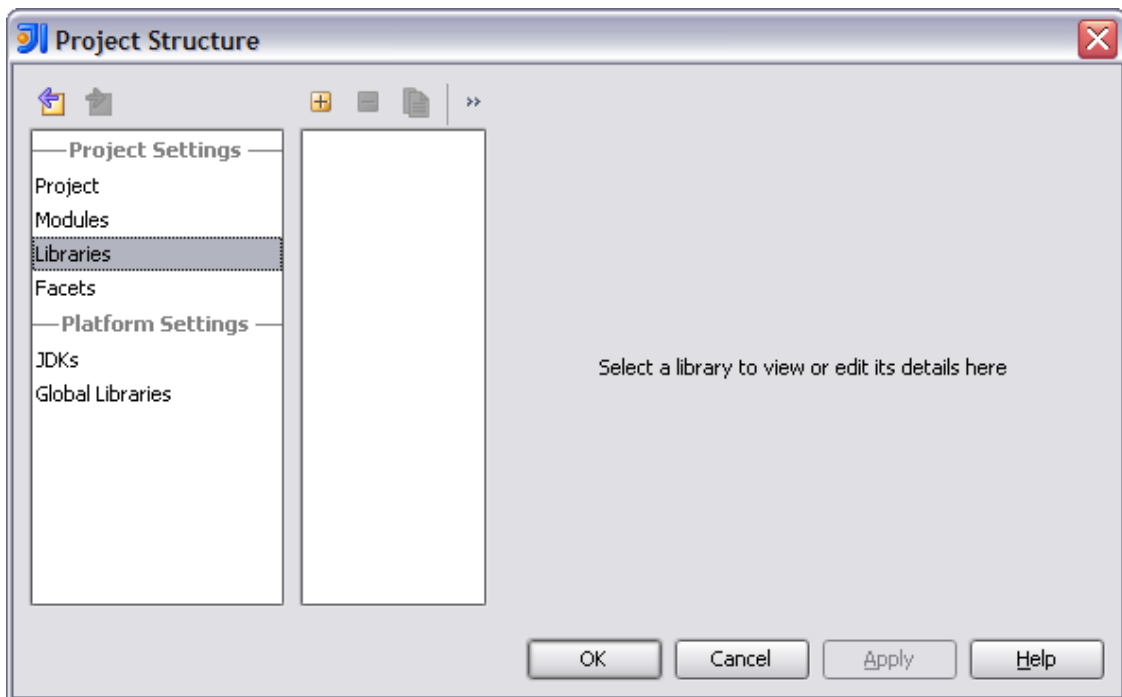
- برای جستجوی سریع فایلها و کلاسها و باز نمودن آنها از **Ctrl + Shift + N** یا **Ctrl + N** استفاده کنید.

- بهتر است در نامگذاری فایلها، پکیجها و کلاسها از فاصله بین کلمات پرهیز کنید. برای نامگذاری کلاسها از روش **PascalCase** استفاده کنید. نام کلاسها را با حرف بزرگ شروع کنید و مابقی را با حرف کوچک ادامه دهید. اگر نام کلاس شامل چندین کلمه است، برای هر کدام از کلمات تشکیل دهنده همین کار را کرده و کلمات را به هم بچسبانید. مثلاً **HelloWorld** یا **ProcessDefinition** و ...
برای نامگذاری پکیجها تماماً از حروف کوچک استفاده نموده، حتی اگر دو کلمه‌ای باشند. مثلاً **helloworld** یا **processdefinition**

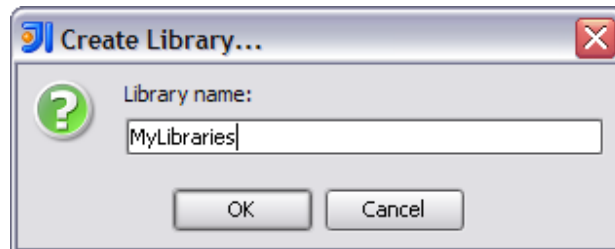
برای نامگذاری متغیرها و object ها از اسامی با معنی و به روش camelCase استفاده کنید. camelCase یعنی اگر متغیر شامل یک کلمه است با حرف کوچک شروع شود (مثلاً process) و اگر چند کلمه‌ایست، کلمه اول با حرف کوچک و مابقی کلمات با حرف بزرگ شروع شوند. مثلاً processDefinition یا stdNumberZero

برای نامگذاری ثوابت تماماً از حروف بزرگ استفاده کنید. برای مثال
`public static final String NONE = "none";`

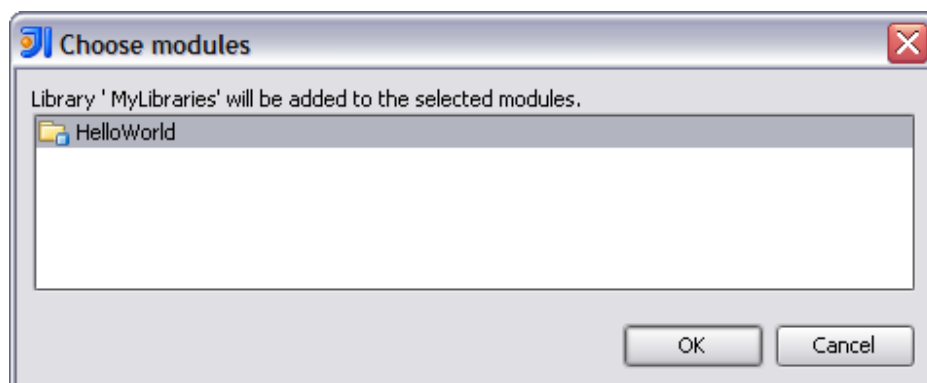
➤ برای اضافه نمودن فایل‌های jar لازم به پروژه، (فایل‌های jar ، library های متفرقه‌ای هستند که شامل مجموعه‌ای از پکیج‌ها و کلاس‌های مرتب می‌باشند و در قالب یک فایل ارائه شده‌اند.) در منوی File گزینه Project Structure را انتخاب کنید تا پنجره زیر باز شود. در پانل سمت چپ آن بر روی گزینه Libraries کلیک نموده و در ادامه بر روی دکمه  کلیک کنید.



در کادر محاوره‌ای زیر نامی (برای نمونه "MyLibraries") وارد نموده و OK را کلیک کنید.



در این پنجره نیز پروژه ای که قصد دارید تا این library به آن افزوده شود را انتخاب کنید. در این مثال پروژه یا ماژول HelloWorld را انتخاب نموده و OK را کلیک کنید.



در این پنجره Attach Classes... را کلیک نموده و فایل jar مورد نظر را در کادر محاوره‌ای Select Path انتخاب کنید. در این مثال فایل jbpm-3.1.2.jar انتخاب شده است. در انتها تمام پنجره‌های باز را OK کنید.

