

1. JAVA

Osnovni pojmovi

Java je platforma za izvršavanje programskog koda i Java je programski jezik. Java kao platforma je dizajnirana tako da što manje zavisi od specifičnih karakteristika konkretnog računarskog sistema, odnosno da jednom napisan i preveden program može da se izvršava na bilo kojoj platformi koja podržava Javu. Dakle, kod napisan u Java programskom jeziku se ne prevodi u izvršni kod (exe) za određeni hardver, nego u izvršni kod (bajt kod – class fajlovi) za Java Virtuelnu Mašinu (JVM). Java virtuelna mašina izvršava bajt kod na platformi na kojoj je instalirana koristeći dostupan hardver.

Java kao programski jezik je jezik opšte namene za objektno-orijentisano programiranje.

Instalacija

Java instalacija dolazi u dva oblika: JDK (Java Development Kit) i JRE (Java Runtime Environment). JRE je podskup od JDK i sadrži samo alate da se izvršava prethodno razvijen i preveden Java kod (dakle, uključuje i Java virtuelnu mašinu - java). JDK pored alata za izvršenje prevedenog Java koda (bajt koda), sadrži i alate za razvoj aplikacija pisanih u Javi (uključuje i Java kompajler – javac).

Provera da li je postoji java na računaru. Otvorite konzolu i ukucajte:

```
java -version
```

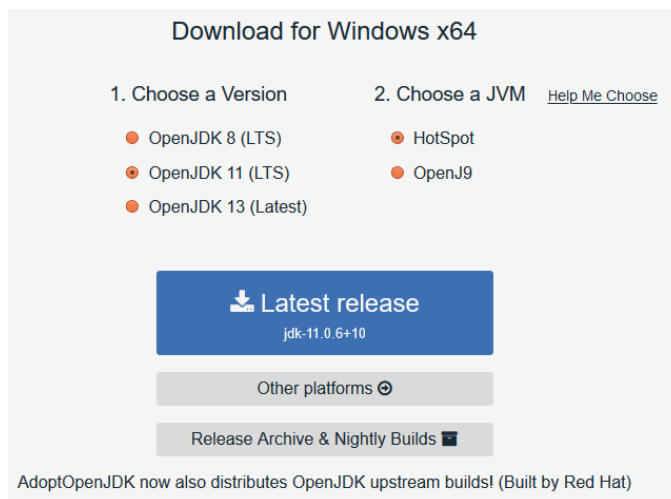
Provera da li je postoji java kompajler na računaru. Otvorite konzolu i ukucajte:

```
javac -version
```

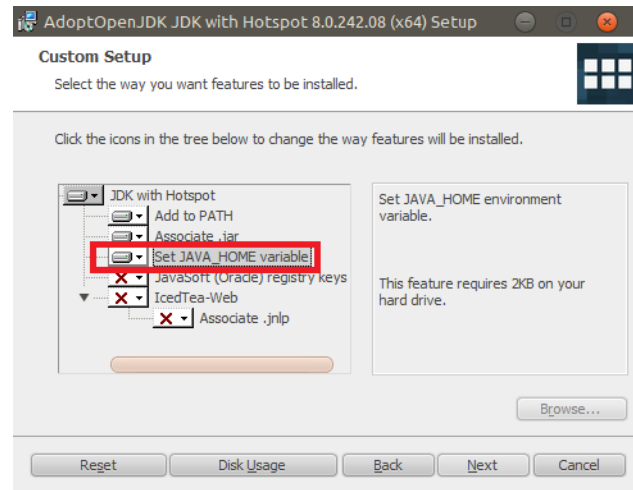
Službena Java implementacija je u vlasništvu firme Oracle i s obzirom na poslednje izmene u licenci za njeno korišćenje, kistićemo OpenJDK verziju Java JDK paketa. Za potrebe ovog predmeta korišćemo JDK verzije 11. U ovom uputstvu će biti pokazano preuzimanje i instalacija JDK paketa sa [AdoptOpenJDK](#) sajta za 64-bitni Windows operativni sistem.

Koraci za preuzimanje i instalaciju su sledeći:

1. Na AdoptOpenJDK [početnoj stranici](#) je potrebno odabrati JDK 11 i HotSpot verziju virtuelne mašine (operativni sistem će automatski biti podešen u odnosu na Vaš trenutni OS, ali je moguće naći i preuzeti verzije za druge sisteme klikom na dugme *Other platforms*):



2. Nakon odabira, Vašu konfigurisanu instalacionu datoteku preuzimate klikom na veliko plavo dugme sa natpisom *Latest release*.
3. Tokom instalacije potrebno je omogućiti opciju „Set JAVA_HOME variable“ u koraku prikazanom na slici ispod (omogućavanje opcije se vrši klikom na nju i odabirom Will be installed form local har drive):



4. Ukoliko nemate opciju postavljanje JAVA_HOME sistemske varijable pogledajte uputstvo za Windows 10 <https://javatutorial.net/set-java-home-windows-10>
5. Nakon ovoga, pokretanjem komandi `java -version` i `javac -version` možete proveriti da li je instalacija uspešno izvršena:

6. Pored ovoga uputstva, službeno AdoptOpenJDK uputstvo za ovu ali iza druge platforme možete dobiti na njihovoj [stranici sa dokumentacijom](#).

Kompajliranje i pokretanje Java koda

Kompajliranje java klasa iz konzole postiže se komandom `javac`. Kao parametri poziva pomenutoj komandi se prosleđuje ime java klase, pri čemu će rezultat biti odgovarajući `.class` fajl (mogu se prosleđivati i drugi parametri). Npr. izvršavanjem komande `javac Test.java` nastaje fajl `Test.class`. Pokretanje java programa unutar Java virtuelne mašina postiže se upotrebom komande `java Test.class`.

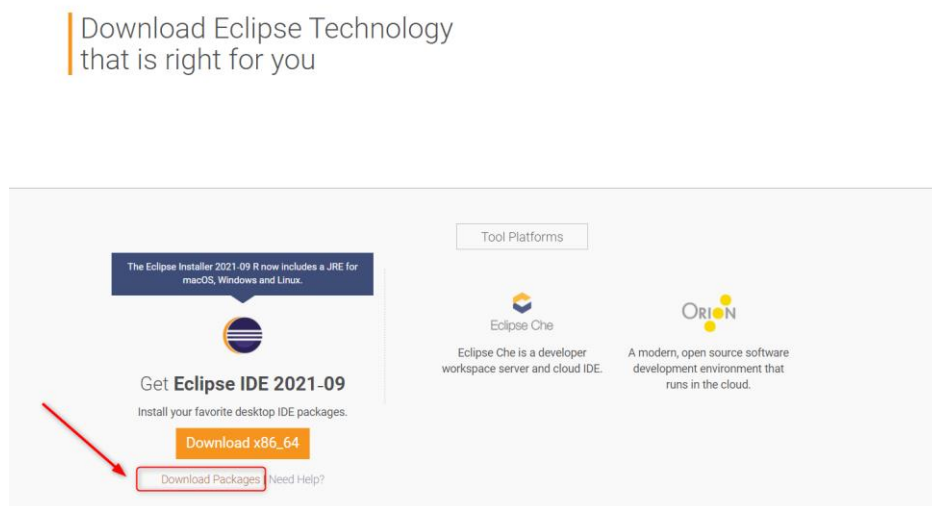
Ukoliko je za kompajliranje ili pokretanje potrebno pristupiti i drugim klasama *javac* i *java* će te klase tražiti upotrebom CLASSPATH varijable. Za kompajliranje java fajlova i njihovo pokretanje na ovom kursu korist ćemo alat Eclipse koji nam ove radnje olakšava.

2. Eclipse

Eclipse predstavlja open-source razvojnu platformu i skup aplikativnih okruženja za izgradnju softvera (nije ograničen samo na Java aplikacije). U ovom momentu je verovatno vodeće razvojno okruženje za razvoj Java aplikacija.

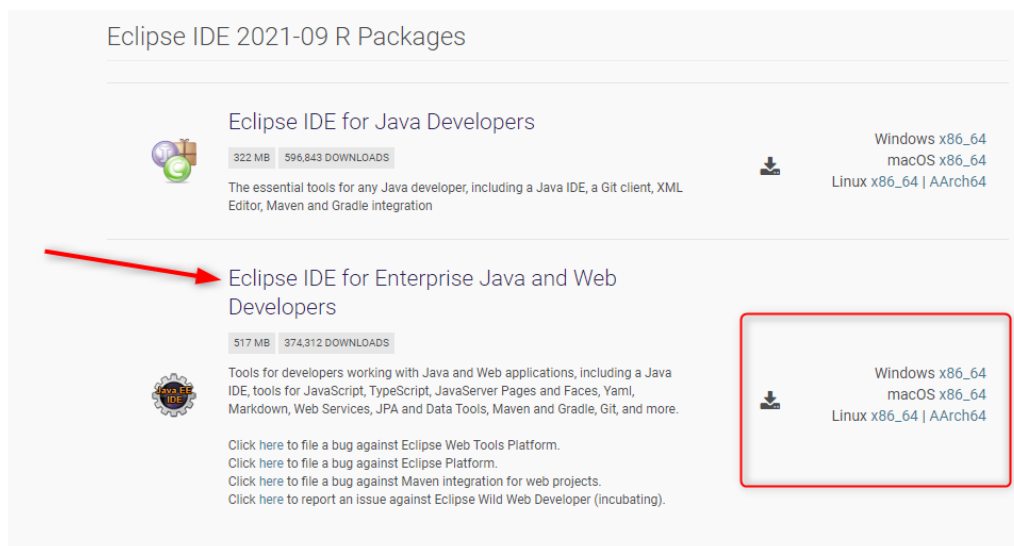
Instalacija

Za Windows platformu je potrebno preuzeti poslednju verziju Eclipse alata sa Eclipse stranice za [preuzimanja](#). Na ovoj stranici, putem linka „Download packages“, možete i preuzeti Eclipse verzije i za druge platforme.



Klikom na linka „Download packages“ otvara se nova stranica na kojoj možete preuzeti različite Eclipse verzije za različite namene.

Neophodno je preuzeti *Eclipse IDE for Enterprise Java and Web Development* verziju koja je prepodešena za razvoj web aplikacija u programskom jeziku Java. Preuzeti automatski podešenu verziju za Vaš operativni sistem.



< ----- Podašavanje i provera Jave za program Eclipse----->

Ukoliko imate jednu verziju JDK paketa instaliranu, Eclipse će automatski biti podešen da je koristi. Međutim, ukoliko je u nekom trenutku potrebno podesiti Eclipse da koristi neki drugi JDK ili JRE to je moguće putem Eclipse menija:

Window → Preferences → Java → Installed JREs

Podešavanje JDK i kompatibilnosti kompajlera u Eclipse programu odlaskom na

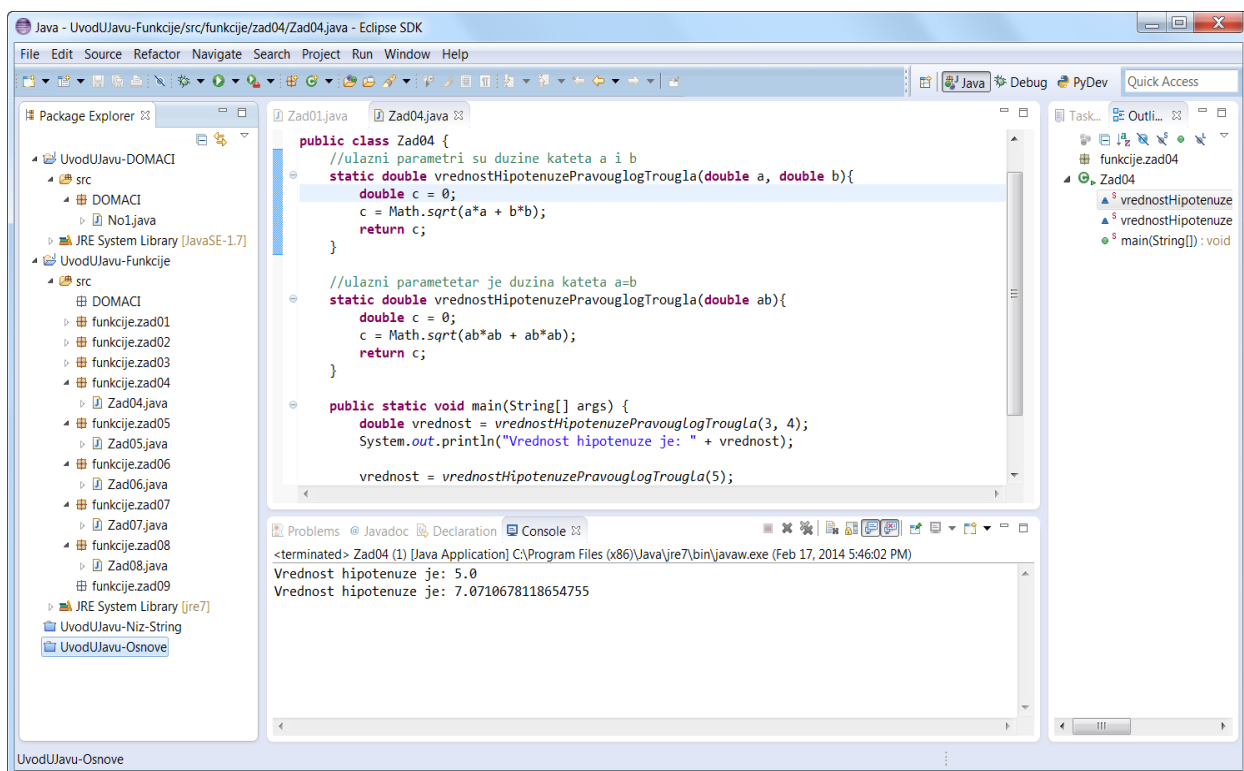
Window → Preferences → Java → Compiler

Kompatibilnost pokretanja i kompajliranja Java programa je bitna ukoliko je kod inicijalno pisan za jednu (npr. java 5) a pokušava se kompajlirati za drugu (npr. 7) verziju jave (različite verzije Jave).

Pri kreiranju novog Java projekta, podešavanje Jave preuzeće se iz podešavanja Eclipse. Podešavanja Jave moguće je naknadno izvršiti u okviru svakog Java projekta zasebno.

Okruženje:

Pokretanjem okruženja otvara se radni prostor (workspace). Za isti eclipse može biti definisano više workspace-ova u kojima su definisana određena korisnikova podešavanja. Radni prostor je moguće organizovati na više načina, odnosno, perspektiva (perspective) koje uključuju određeni skup alata i specijalizovane su za različite namene (desni gornji ugao eclipse alata; perspektive: Java - razvoj standardnih java aplikacija, Java EE - prošireno standardno okruženje koje se npr. može koristiti za razvoj web aplikacija, PyDev - razvoj python aplikacija, Debug - okruženje za debugovanje/traganje grešaka).



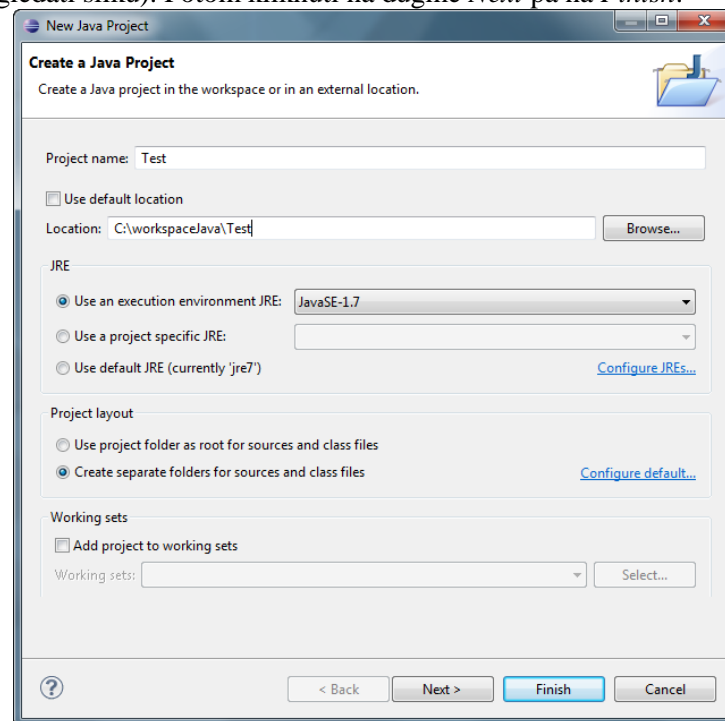
Radni prostor Java perspektive sastoji se od pogleda: **PackageExplorer** (prikazuje listu projekata koji se nalaze u radnom prostoru i hijerarhiju Java resursa u okviru projekata), **Outline** (prikazuje listu atributa i metoda klase koja se edituje), **Problems** (prikazuje tabelu grešaka i upozorenja vezanih za izvorni kod), **Console** (prikaz poruka na ekran za pokrenuti java program) i **Editor** (zauzima centralni deo površine radnog prostora). Pogledi koji se nalaze u određenoj perspektivi se mogu prilagoditi potrebama i

navikama programera. Mogu se dodavati novi ili brisati postojeći pogledi iz perspektiva, a mogu se i menjati veličine i položaj pogleda unutar radnog prostora. Novi pogledi se dodaju iz opcije glavnog menija *Window->Show view*.

Rad sa projektima

Programski kod se organizuje unutar projekata. Projekat se sastoji od hijerarhije klasa (u skladu sa pravilima definisanja paketa u Java programskom jeziku) i drugih datoteka (npr. slika, txt fajlova, konfiguracionih datoteke itd.). U okviru radnog prostora može da bude definisano više projekata. Projekat može da bude otvoren (ikonica otvorena fascikla) ili zatvoren (ikonica plava zatvorena fascikla). Samo otvoreni projekti su aktivni, tj. nad njima je jedino dozvoljen rad u eklipsi. Zatvoreni projekti ne mogu da se menjaju jer nisu aktivni (ali se i dalje nalaze u lokalnom fajl sistemu). Poželjno je zatvarati nepotrebne projekte jer se na taj način oslobađaju zauzeti memorijski resursi i podižu performanse Eclipse okruženja.

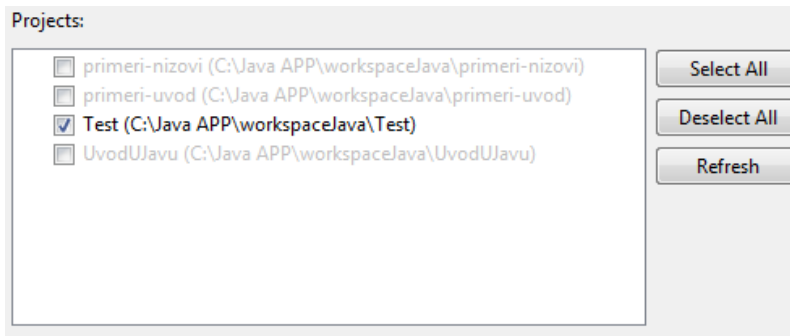
Kreirati svoj prvi java projekat pokretanjem vizarda *File→New→Java Project*. Dodeliti vrednost *Test* kao ime projekta (pogledati sliku). Potom kliknuti na dugme *Next* pa na *Finish*.



Zatvaranje otvorenog projekta - Desni klik na projekat pa stavka **Close Project**. Projekat je zatvoren ali ne i uklonjen iz okruženja.

Uklanjanje projekta iz okruženja - Projekte je moguće ukloniti logički i fizički iz radnog prostora. Uklanjanje se postiže desnim klikom na selektovani projekat pa stavka **Delete**.

Otvaranje postojećeg projekta - Prebaciti folder postojećeg projekta u radni direktorijum *WorkspaceJava*. Pozvati opciju *File→Import→General→ Existing project into Workspace* i kliknuti na dugme *Next*. Na dugmetu *Browse* postaviti putanju do radnog direktorijuma *WorkspaceJava*. Iz liste ponuđenih projekata odabrati onaj koji se importuje. Klik na dugme *Finish*.

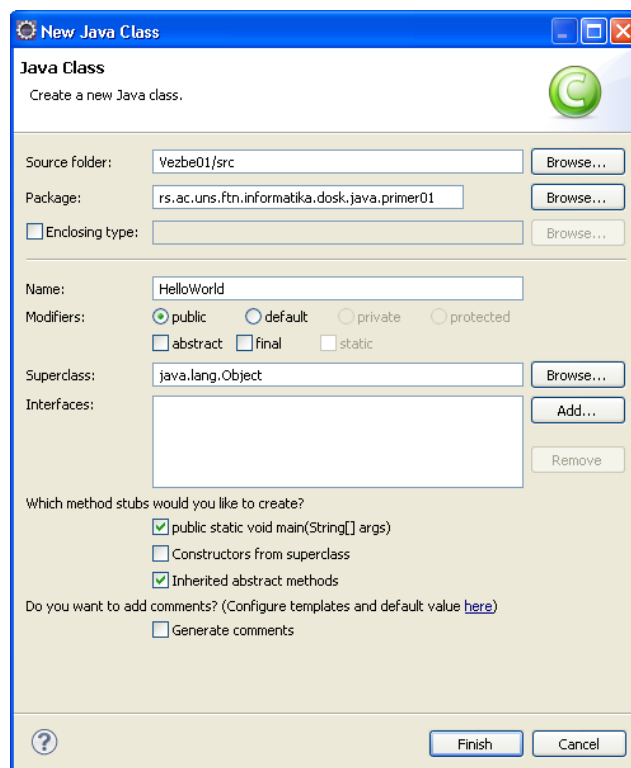


Rad sa paketima

Paketi predstavljaju hijerarhijsku organizaciju fajlova po folderima i podfolderima. Paketi se kreiraju pokretanjem vizarda *desni klik na src folder* → *New* → *Package* i zadavanjem njegovog imena. Kreirati nekoliko novih paketa pod nazivom *vezbe1.podpaket*, *vezbe2*, *vezbe3.podpaket*. Kroz operativni sistem otići na lokaciju projekta i istražiti sadržaj foldera *src*. U pakete se dodaju Java klase. Moguće je promeniti naziv paketa, kopirati klase iz jednog paketa u drugi ili obrisati klase. Imena paketa treba da budu jedinstvena ne na nivou projekta nego na globalnom nivou. Pored paketa u projekte se mogu dodavati i folderi, tekstualni fajlovi, klase, itd.

Rad sa Java klasama

Kreirati svoju prvu Java klasu pokretanjem vizarda *desni klik na paket* → *New* → *Class*. Dodeliti vrednost *HelloWorld* kao ime klase (pogledati sliku) i selektovati opciju *public static void main*. Potom kliknuti na dugme *Finish*.



Nakon toga u centralnom editoru otvara se *HelloWorld* klasa u kojoj se nalazi i *public static void main* metoda. Metoda *main* je mesto odakle počinje izvršenje programa.

Eclipse nudi i autocomplete, odnosno omogućava Vam da unesete deo naredbe i pritisnete Ctrl+Space nakon čega je ta naredba završena. Na primer: unutar main metode napišite syso i pritisnite Ctrl+Space i Eclipse će Vam napisati *System.out.println()*; Unutar zagrada napišite "Hello world!".

Kompajliranje i pokretanje

Kompajliranje se obavlja automatski, ali se ovakvo ponašanje Eclipse-a može i promeniti putem opcije iz glavnog menija *Project->Build automaticaly*. Ako se ova opcija isključi postaju aktivne opcije *Build All* i *Build Project*. Podrazumevano ponašanje Eclipse-a je da kreira *bin* folder u projektu u kojem smešta class fajlove koji se dobijaju kompajliranjem. Ovaj *bin* folder se ne vidi iz *Package Explorer* pogleda, ali se vidi iz proizvoljnog file browser-a, a takođe se može videti i iz Eclipse okruženja putem *Navigator* pogleda (*Window->Show view*).

Pokretanje programa se bira iz padajuće liste koja se dobija desnim klikom na klasu u *Package Explorer* pogleda. U padajućoj listi postoji opcija *Run As -> Java Application* čijom selekcijom se vrši pokretanje Java virtuelne mašine i u njoj izvršenje *main* metode selektovane klase. U projektu može postojati proizvoljno mnogo klasa koje imaju *main* metodu, ali se pokreće *main* metoda selektovane klase (one na kojoj je urađen desni klik). Prečica za ovu akciju je Alt+Shift+X, nakon čega je potrebno pritisnuti J. Nakon ovoga se izvršava *main* metoda selektovane klase. Eclipse ima mnogo prečica, a njihova lista se može videti (a i menjati) putem opcije *Window -> Preferences -> General -> Keys*.

Metoda *main* može imati i listu ulaznih parametara koji se mogu proslediti prilikom pokretanja programa putem opcije iz padajuće liste koja se dobija desnim klikom na klasu u *Package Explorer* pogleda: *Run As -> Run Configurations...*

Debug

Eclipse okruženje poseduje posebnu perspektivu za debug-ovanje aplikacije (otkrivanje grešaka u aplikaciji) koja pruža mnoštvo mogućnosti. Postoji mnogo dostupnih tekstova o debug-ovanju Java koda pomoću Eclipse-a kao što je na primer: <http://www.vogella.com/tutorials/EclipseDebugging/article.html>, ili materijal Eclipse – debug.pdf koji se takođe nalazi među materijalima ovog kursa.

Plug-ins

Eclipse okruženje je otvorenog koda ali i otvorene arhitekture i vrlo ga je lako proširiti. Postoji mnoštvo proširenja za Eclipse, a neke od njih ćete i raditi tokom ovog kursa (Subclipse). Instaliranje novog dodatka se može postići opcijom glavnog menija *Help->Install New Software...* (to nije jedini način za instaliranje, ali je najčešći). Potreban Vam je *update site* nekog dodatka (Na primer: http://subclipse.tigris.org/update_1.10.x). Postoje i sajtovi koji sadrže listu dodataka za Eclipse kao što je: <http://marketplace.eclipse.org/>.