**JAVA EE**

🡺 Viacvrstvova arhitektura

🡺 J2EE vrstva – prezentačná & business logika

🡺 Klient <> server

🡺 Vygenerovanie HTML na server strane

🡺 Business logika na server side 🡺 databazy...

🡺 Enterprise appky – distribuovane, viacvrstove, **platform independent (server)**...

🡺 **EJB – SERVER SIDE COMPONENTS (dnes)**

🡺 Frontend – servlety...

🡺 JEE – XMLka

🡺 Classa na app. serveri – služba s nejakými metódami – viem to volať

🡺 EJB Container, Web cont., Support services – cachce, security (BUILT IN) 😊...

🡺 APP SRV. – Apache, Oracle, RedHat, IBM, JBoss... – lisia sa (podpora JEE)

🡺 Pozor, aby sme sa „nezaviazali“ na nejake proprietarne API (podpora...)

🡺 Common tasks – remote access, concurency, persistence, transactions... (ocakavam to od servera)

🡺 Cielom je sustredit sa na to **CO TO BUDE ROBIT, NE AKO 😊**

🡺 Bean provider – to som ja (biznis logika) – bean prov. 🡺 app. assembler 🡺 deployer 🡺 system administrator

🡺 Administrator – monitoring utilizacie...

**EJB**

🡺 Definuju sposob, ako vyvijat komponentovo-orientovane a distribuovane aplikacie

🡺 **Server-side komponenty, pomocou ktorych implementujeme biznis logiku**

🡺 Komponent = classa; klient vykonáva metódy

🡺 EJB specifikacia 🡺 komunikacia ja <> server

🡺 Platform independent

🡺 EJB server ~ app. container

🡺 When to use – poistovanie aut 🡺 bezpecnost, transakcie...

🡺 EJB robia tu „tazku pracu“ (biznis logiku)

**EJB ARCHITECTURE:**

🡺 Beaning v správe app. servera

🡺 Beany (EJB) – stateful, stateless (session beans)

🡺 JPA – JavaPersistenceAPI – aj to je EJB (db)

🡺 **Session bean – classa s metódami biznis logiky**

🡺 Sú krátko žijúce – žijú v pamäti app. servera; nezdieľajú sa medzi klientami (ani stavy)

🡺 Klient aj server potrebujú remote rozhrane

🡺 **Na strane klienta metódy – STUBY, servera – SKELETONY NA SKUSKU !!!!!!!!!**

🡺 EJB volam odinakial – remote volanie, z lokalu – stačí lokálny pohľad

🡺 Anotácie v EJB 3

🡺 **Deployment descriptor** – „anotácie“ vytiahnutú do XML – ľahko to prispôsobím napríklad pre inú db – v´dôležité, lebo je to **flexibilné**

**J2EE PRAX**

🡺 Zapnem J2E v Eclipse – Servers

🡺 New Server 🡺 otvorim stiahnuty JBoss (alebo daco ine - **WildFly**) – POZOR NA KOMPATIBILITU!!!

🡺 Local, .. and shell.. 🡺 home dir (vyberem app. srv.) – jre 1.8

🡺 Adresár standalone 🡺 deployments

🡺 Logy

🡺 New project 🡺 EJB 🡺EJB Project

🡺 **MAT J2EE ECLIPSE!!!!!!**

🡺 Novy folder (src) 🡺 New session bean TestBean 🡺 Remote rozhranie

🡺 Add projekt to server

🡺 Aky server + aky port – CRUCIAL

🡺 Parametre servra sa mozu lisit v zavislosti od verzie ☹ -- BACHA NA TO !

🡺 Závislosť medzi projektami (klient <> server) 🡺 Java build path (len kvoli importom pre jednoduchost, inak je to hovadina)

🡺 jboss-cleint JARko – aby to išlo -\_- (z klienta na 1 PC zavolam server) – samozrejme realne robim to vzdialene

🡺 Da sa to aj debugovať cez breakpointy

🡺 Vedieť čo je app. server, komponentová architektúra, b. logika, EJBčka (EJE – BEČKA), seesion beany – **stateless** & stateful

🡺 O tie Jarove kody sa oprieť ked budeme

🡺 **Skusit si to rychlo rozbehat teraz, aby nas to potom nesokovalo (ANO, TYKA SA TO AJ NAS) – KAZDY MAME SERVER!!!**

**🡺 CIZE SI NAPISAT PRVE EJBCKO A ZISTIT, CI TO SLAPE**