

# Portal

## Mi a portál?

Általánosságban portálnak nevezünk olyan webalkalmazásokat, amelyek tematikus tartalmat gyűjtenek, illetve a felhasználók és fleg a szerkesztők számára lehetővé teszik a tartalmak szerkesztését, illetve a portál testreszabását. A JSR-168 specifikáció kissé eltér portál képet fogalmaz meg, de ezt nagy vonalakban megfelel a portálokról kialakult képnek.

A JSR-168 egyik nagy hibája, hogy *nagyban gondolkodik*, s bár nem része a Java Enterprise Edition családnak: sok közös pontra lehet lőni. Egy Java portál éppen ezért inkább nagyobb cégeknél volt jellemző, s ott is inkább csak intranetre szánt program volt. Kevés olyan JSR-168 portált látni, amely kisebb cégekre szabható, illetve megfelel a Web2.0 kívánalmaknak és egyben kitehet az internet nyilvánossága elé. Még kevesebb olyan megoldást találni, amely egyidejűleg - egy példányt feltételezve - több különböző domént kezel, és ehhez azonos portlet példányokat használ, közös a portlet konténer... de erről majd később... ☺

Tipikus példája a portálnak a <http://www.javaforum.hu> portál:



A portálok általában oldalakat, illetve oldal-hierarchiákat tartalmaznak, s ezek az oldalak hordozzák a portleteket. Egy portlet több oldalon is szerepelhet.

## Mi a JavaForum 2.0 portál célja?

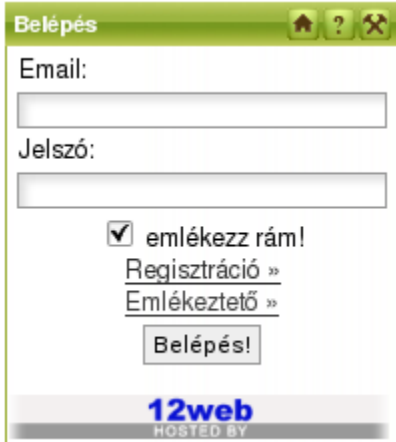
A JavaForum 2.0 olyan Java alapú portál motor, amely teljesíti a JSR-168 (illetve idvel a JSR-286) specifikáció előírásait, s ezzel együtt egy pár lépéssel telepíthető és könnyen üzemeltethető portált ad a felhasználók kezébe, amelyet eddig csak a népszerűbb PHP portál motoroktól láttunk.

## Mi az a portlet?

A portlet fogalmát a Sun hozta létre (mint az applet és a servlet fogalmát is), jelentése kikövetkeztethető: a portál egy kisebb (önálló) része. A portlet egy nagyon pontosan specifikált csomagban érkezik, és ezt a csomagot *bármelyik* Java portálnak tudnia **kell** telepíteni és üzemeltetni.

Egy portlet úgy kell viselkedjen, mintha egyedül lenne, nem kell tudnia arról, hogy a portál melyik részén, illetve milyen környezetben jelenik meg. Egyszeren fogalmazva a portlet lát egy kérést, abból kiveszi a neki szóló paramétereket, csinál valamit a paraméterek alapján, majd a válaszból írja azt, amit tartalomnak elállít. Természetesen ennél összetettebb a dolog, hiszen a portleteknek vannak módjai (VIEW, EDIT és HELP), a portlet ablaknak vannak méretei (MINIMIZED, NORMAL és MAXIMIZED), illetve kezelhet jogosultságot, és végezhet akciót.

Tipikus példája a portletnek a <http://www.javaforum.hu> portál belépést kezel portlete:



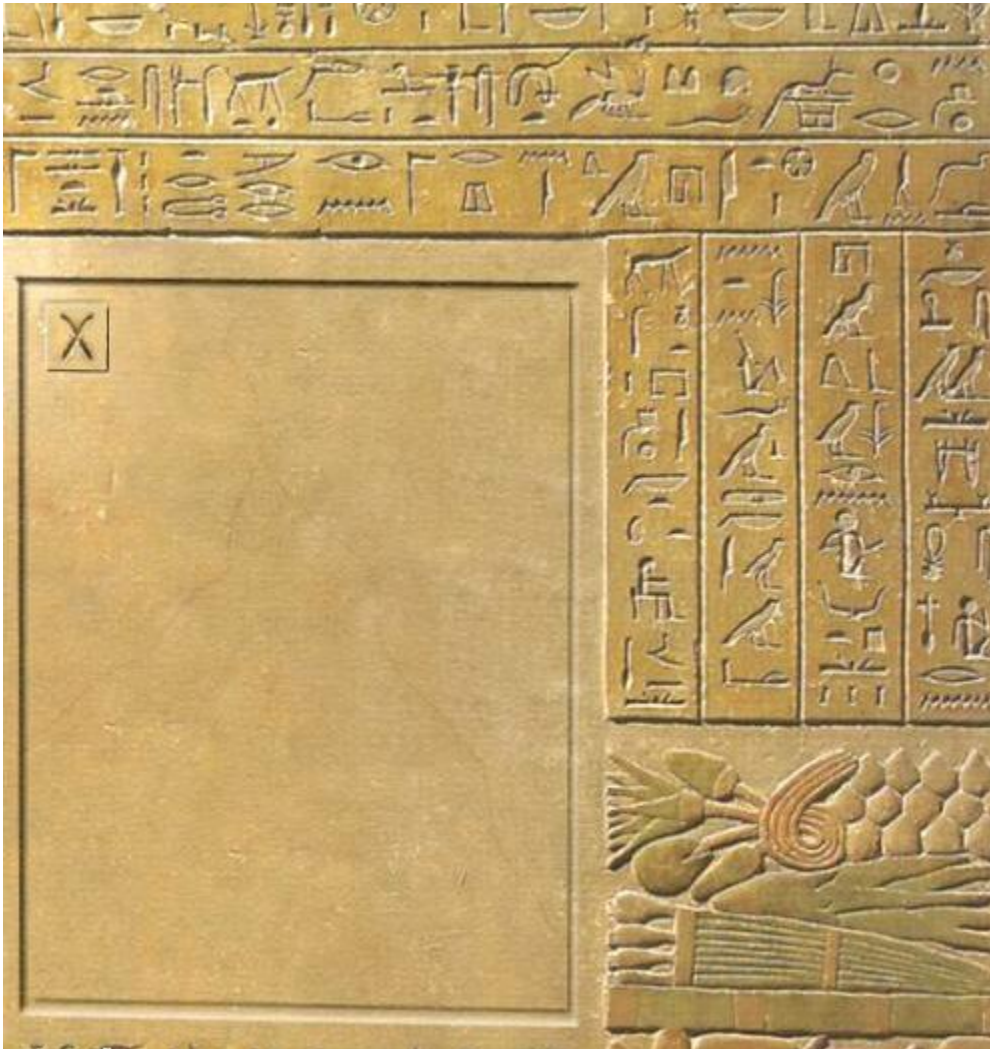
A portál a portletek kimenetét felhasználhatja bármelyik oldalon, ezért minden egyes portletnek fontos a kimenetét úgy elállítania, hogy ezzel ne rontsa el a portál által összerakott kimenetet, amely általában több portlet kimenetét tartalmazza egybe csomagolva.

## A portlet konténer

A portletek kezeléséért a portlet konténer felel, elindítja és leállítja azokat, amikor erre szükség van. A portletek egy sor olyan utasítást adhatnak ki (főképp az akciók feldolgozásánál), amelyekkel a portlet konténert utasítják: portlet jellemző mentése, portlet állapot váltása, stb.

## A portál konténer

A portál konténer feladata sokkal összetettebb, mint a portlet konténeré, mivel a portál fogadja az összes kérést, azokat fel kell dolgoznia, a feldolgozott adatokból ki kell nyernie a kért oldalt, jogosultságot kell kezelnie, az oldalhoz tartozó portleteket meghatározni, a portleteknek szánt paramétereket kinyerni a kapott URL-ből, majd a portlet konténert használva meg kell hívni minden egyes portletet, átadva a neki szóló paramétereket, a portletek választását pedig bele kell illeszteni az oldal template-be.



[<< Vissza](#)