

Stručný obsah

Úvod	7
1. část: Základy JSP	
1. Úvod do internetových technologií	14
2. JSP Elementy: komentáře a direktivy	47
3. JSP elementy: skriptovací elementy	73
4. Implicitní objekty	98
5. Standardní akce	145
6. Knihovny značek	179
2. část: Sestavování reálných aplikací	
7. Architektury JSP	224
8. JSP a bezpečnost	253
9. Kontroler a řízení toku ve webových aplikacích	290
10. Používáme databázi	323
3. část: Dodatky	
Příloha A: Odpovědi na otázky v závěrečných testech	362
Příloha B: Vytvoření vývojového a produkčního prostředí	380
Příloha C: Vytvoření MySQL databáze	395
Příloha D: Nové vlastnosti JSP 1.2	402
Rejstřík	407

1. Úvod do internetových technologií

1.1 Internet jako architektura klient-server	15
Problémy s transparentností sítě	16
Stavové servery	17
HTTP servery jsou bezstavové	18
1.2 HTTP požadavek	19
Řádek požadavku	19
Metody požadavku	19
URI adresy v požadavku	21
Verze HTTP protokolu	23
Záhlaví zprávy	23
Tělo zprávy	23
1.3 HTTP odezvy	24
Stavový řádek	24
Verze HTTP protokolu	24
Stavový kód a popis příčiny	24
Záhlaví zpráv	25
Tělo zprávy	26
1.4 Ostatní součásti	26
Projekt 1.1: Ozvěny HTTP protokolu	26
1.5 Programování webových aplikací na straně klienta	31
Rozšíření webových prohlížečů	32
Pomocné aplikace	33
Zásuvné moduly prohlížeče	33
JavaScript	33
VBScript	34
Dynamické HTML	34
Java aplety	35
Úloha a omezení technologií prohlížečů	36
1.6 Webové programování na straně serveru	37
Rozšiřování webového serveru	40
Serverový JavaScript	40
Serverové zásuvné moduly	40
ASP	41
Java servlety	42
JSP	43
Aplicační servery Java 2 Enterprise Edition (J2EE)	43
Úloha Java servletů a JSP stránek	44
Projekt 1.2: Spuštění prvního JSP programu	45
1.7 Závěrečný test	46

9. Kontroler a řízení toku ve webových aplikacích

9.1 Kontroler.....	291
Zpracování vstupu uživatele	291
9.2 Spolupráce s modelem	293
9.3 Řízení toku webových aplikací.....	293
9.4 Jednoduché řízení toku.....	301
Projekt 9-1: Vyhledávání v adresáři	306
9.5 Pokročilé řízení toku: Struts MVC framework.....	318
9.6 Závěr	320
9.7 Závěrečný test.....	322

Cíle kapitoly:

- ☞ porozumět významu kontroleru v architektuře MVC
- ☞ porozumět řízení běhu ve webových aplikacích
- ☞ naučit se postupy pro vytváření struktur s konfigurovatelným řízením toku

Technologie JSP pomáhá usnadňovat vývoj dynamických webových aplikací, ale webové aplikace se přesto velmi rychle stávají komplexními. Pokud vytváříte složitou aplikaci, měli byste nejprve vytvořit dobrou architekturu s využitím ověřených technik pro dosažení flexibility a rozšiřitelnosti. Dobrá architektura je taková, která pomůže aplikaci přestat zkoušku času.

V kapitole 6 jsme si ukázali, jak lze JSP stránky zjednodušit pomocí knihoven značek (*custom tag libraries*). Ty pomáhají odstranit mixování Java kódu a HTML. Knihovny značek činí JSP stránky čitelnějšími a adaptabilnějšími. V kapitole 7 jsme se zabývali různými návrhovými vzory pro webové aplikace a ukázali jsme si význam vzoru *Model-View-Controller* (MVC) pro architektury webových aplikací. Přestože je využití knihoven značek pro zjednodušení JSP stránek (představující *view* v architektuře MVC) důležité, nepředstavuje to úplnou architekturu MVC. V této kapitole se zaměříme na MVC *kontroler*, který koordinuje komunikaci mezi *modelem* a *view*.

10. Používáme databázi

10.1 Role externích zdrojů dat	324
Správa rozsáhlých dat.....	325
Centralizace, zabezpečení a sdílení dat	325
Poskytnutí dynamických dat.....	326
Přístup ke starším informačním systémům	327
10.2 JDBC API	329
Architektura JDBC	329
Základy JDBC	330
Služby pro připojení	330
SQL služby.....	333
Zpracování výsledků	335
JSP a rozhraní JDBC	339
Knihovna značek projektu Jakarta pro práci s databází	339
Kdy nepoužít knihovnu značek pro práci s databází	340
10.3 Datový model Rodinného centra	344
Typy entit	345
Atributy	347
Vazby	347
10.4 Způsob realizace přístupu k datům	351
Vzor Most.....	351
Použití databáze pro implementaci naší chráněné oblasti	351
Model pro Adresář	357
10.5 Závěrečný test	360

Cíle kapitoly:

- ☞ ukázat si roli externích zdrojů dat v reálných aplikacích
- ☞ porozumět významu databází v JSP aplikacích
- ☞ ukázat si základy rozhraní JDBC pro práci s daty
- ☞ ilustrovat použití relační databáze v naší aplikaci Rodinné centrum

V kapitole 8 jsme si ukázali různé aspekty bezpečnosti webových architektur. Popsali jsme si také přihlašovací stránku naší aplikace Rodinné centrum. V kapitole 9 jsme diskutovali o řízení toku JSP aplikací. Také jsme si popsali kontroler pro aplikaci Rodinné centrum. Nyní si v této poslední kapitole ukážeme, jakým způsobem přidat do JSP architektury *externí zdroj dat* (např. relační databázi) a popíšeme databázovou implementaci bezpečnosti a správu dat v naší aplikaci a to pomocí JDBC.

Příloha A:

Odpovědi na otázky v závěrečných testech

A.1 Úvod do internetových technologií	362
A.2 JSP Elementy: komentáře a direktivy	363
A.3 JSP elementy: skriptovací elementy	364
A.4 Implicitní objekty	366
A.5 Standardní akce	368
A.6 Knihovny značek	371
A.7 Architektury JSP	373
A.8 JSP a bezpečnost	375
A.9 Kontroler a řízení toku ve webových aplikacích	377
A.10 Používáme databázi	378

A.1 Úvod do internetových technologií

1. Co znamená HTTP?

HTTP je zkratkou pro *Hypertext Transfer Protocol*.

2. Uveďte tři rysy počítačové sítě, kvůli kterým je obtížné dosáhnout její transparentnosti.

☞ **Zpoždění:** Časová prodleva, než je požadavek předán serveru a vrácena odezva.

☞ **Spolehlivost:** Pravděpodobnost odezvy na požadavky klienta.

☞ **Souběh:** Pořadí zpracování požadavků na straně serveru.

3. Napište dva typy HTTP požadavků.

Dva nejdůležitější typy HTTP požadavků jsou *GET* a *POST*. *GET* získává určitý webový prostředek ze vzdáleného serveru, přičemž posílá informace spolu s požadovanou URI. *POST* se používá k posílání a získávání dat ze vzdáleného serveru. Narozdíl od *GET* přikládá *POST* data na konec požadavku jako zvláštní obsahovou zprávu.

2. JSP Elementy: komentáře a direktivy

2.1 JSP kontejner	48
Fáze JSP	48
JSP rozsah.....	49
JSP konstrukce.....	51
Pohled na celou JSP stránku z pačičí perspektivy	52
2.2 JSP komentáře	53
Použití komentářů.....	53
Typy JSP komentářů.....	55
Komentáře zdrojového kódu JSP stránky.....	55
Komentáře HTML dokumentu	55
2.3 JSP direktivy	56
Direktiva include	57
Relativní URL	57
Použití direktivy include	58
Omezení direktivy include	59
Projekt 2.1: Test direktivy include#	61
Direktiva page	62
Atributy direktivy page	63
Projekt 2.2: Test direktivy page#	69
Direktiva taglib.....	69
2.4 Závěrečný test	71

Cíle kapitoly:

- ☞ porozumět JSP kontejneru
- ☞ pochopit jakým způsobem se mají zapisovat komentáře do JSP kódu
- ☞ poznat základní JSP direktivy

JSP stránka se podobá stránce v jazyce HTML, obsahuje ale zvláštní syntaktické elementy, díky kterým může stránka dynamicky měnit svůj obsah. V této a v následujících čtyřech kapitolách jsou tyto elementy vysvětleny.

Zpracování JSP stránek probíhá rozdílně od statických HTML dokumentů. Během jednoduché komunikace mezi webovým serverem a prohlížečem je požadovaný soubor přenesen jako datový tok ze serveru k prohlížeči, který jej převede na obrazovou informaci a zobrazí ve svém okně. Je-li požadovaný soubor statický HTML dokument, vloží

3. JSP elementy: skriptovací elementy

3.1 Tři skriptovací elementy.....	74
3.2 Jak kontejner překládá skriptovací JSP elementy	74
3.3 JSP deklarace.....	78
Proměnné v JSP deklaraci	78
Metody v JSP deklaraci	79
Deklarace složitějších vlastností Javy	79
Další informace o JSP deklaracích	80
3.4 JSP výrazy.....	83
Další informace o JSP výrazech	84
3.5 JSP skriptlety	85
Proměnné ve skriptletech.....	85
Výrazy ve skriptletech	86
Podmínky ve skriptletech	87
Cykly ve skriptletech	88
Složitější vlastnosti Javy ve skriptletech	89
Lokální vnitřní třídy ve skriptletech	89
Anonymní třídy ve skriptletech	90
Výjimky ve skriptletech	90
Další informace o JSP skriptletech	91
Projekt 3.1: Skriptovací elementy#.....	92
3.6 Závěrečný test.....	97

Cíle kapitoly:

- ☞ naučit se různé skriptovací elementy
- ☞ pochopit, jak kontejner skriptovací elementy překládá
- ☞ seznámit se s omezeními pro použití skriptovacích elementů

V kapitole 2 jste viděli, jak se JSP komentáře používají k objasnění záměrů programátora a jak JSP direktivy pomáhají konfigurovat JSP kontejner. Nyní se naučíte skriptovací JSP elementy, které vám umožní vkládat přímo do JSP stránek kód Javy. Možnost přímého vložení Javy do JSP stránky byla nejdůležitějším faktorem, který zapříčinil velkou oblibu programování JSP.

4. Implicitní objekty

4.1 Proměnná request	101
Služby proměnné request	101
Přístup k parametrům požadavku	101
Správa atributů v rozsahu požadavku	104
Předávání cookies	105
Přístup k záhlavím požadavku	105
Přístup k elementům řádku s požadavkem	106
Přístup k informacím o zabezpečení	107
Přístup k informacím o mezinárodním nastavení	108
4.2 Proměnná response	109
Služby objektu response	110
Ukládání výstupu do vyrovnávací paměti	110
Nastavení záhlaví odezvy	112
Přesměrování prostředků	114
Projekt 4.1: Odesílání datového toku s obsahem souboru#	115
4.3 Proměnná pageContext	118
Služby proměnné pageContext	119
Přístup k více rozsahům	119
Přístup k implicitním objektům	120
Správa vnořených rozsahů	121
Předávání a vkládání	122
4.4 Proměnná session	123
Přepisování URL	123
Instalace Cookies	123
Relace SSL	124
Služby proměnné session	124
Vázání atributů	124
Správa časových omezení relace	125
Projekt 4.2: Přesměrování a vkládání URL#	126
4.5 Proměnná out	128
Služby proměnné out	129
Správa vyrovnávací paměti odezvy	129
Zápis obsahu	130
4.6 Proměnná application	133
Služby proměnné application	133
Přístup k inicializačním parametrům	134
Správa atributů kontextu aplikace	134
Podpora abstrakce prostředků	135
Metody objektu RequestDispatcher	136
Metody pomůcek	139
4.7 Proměnná config	141

4.8 Proměnná page.....	142
Služby proměnné page	142
Služby dohod.....	143
Přístup k informacím o servletu	143
4.9 Proměnná exception	143
Služby proměnné exception.....	143
4.10 Závěrečný test.....	144

Cíle kapitoly:

- ☞ seznámit se s různými implicitními objekty
- ☞ pochopit třídy Javy, které jsou implicitními objekty reprezentovány
- ☞ porozumět způsobu používání implicitních objektů na JSP stránkách

Jednou z výhod psaní webových aplikací v rámci Java servletů a JSP kontejnerů je to, že kontejnery poskytují aplikaci služby. JSP stránky tyto služby zpřístupňují prostřednictvím implicitních objektů, které mohou vývojáři v JSP používat.




Specifikace *JSP 1.1* po skriptovacích jazycích JSP vyžaduje, aby umožnily používat společnou sadu **implicitních** objektů. Tyto objekty jsou implicitní v tom smyslu, že je nemusíte před jejich použitím deklarovat. O implicitních objektech jsme se několikrát zmínili v kapitolách 2 a 3, například v kapitole 3 jsme popsali, jakým způsobem mohou JSP výrazy a skriptlety používat *JspWriter* objekt *out*. V této kapitole popisujeme podrobně všechny implicitní objekty a způsoby jejich použití.

V *JSP 1.1*, kde je jediným skriptovacím jazykem Java, odpovídá každý implicitní objekt určité třídě nebo rozhraní Javy. V našem popisu neklademe příliš velký důraz na rozlišení mezi názvy proměnných a typy objektů. Tabulka 4.1 uvádí seznam implicitních objektů a příslušných typů Javy.

5. Standardní akce

5.1 JSP akce	146
Standardní akce a vlastní akce	146
Atributy značky	147
Atribut id	148
Atribut scope	149
5.2 Akce <jsp:useBean>	150
Atributy a pravidla jejich používání	151
Kroky zpracování	151
Použití akce <jsp:useBean>	154
5.3 Akce <jsp:setProperty>	156
Atributy a pravidla používání	157
Dvě varianty akce <jsp:setProperty>	158
Přiřazení hodnot vlastností pomocí parametrů HTTP požadavku	158
Přiřazení hodnot vlastností pomocí textových konstant a JSP výrazů	160
5.4 Akce <jsp:getProperty>	161
Atributy a pravidla používání	161
Použití akce <jsp:getProperty>	162
5.5 Akce <jsp:include>	163
Atributy a pravidla používání	164
Použití akce <jsp:include>	164
5.6 Akce <jsp:forward>	165
Atributy a pravidla používání	166
Použití akce <jsp:forward>	166
5.7 Akce <jsp:param>	167
Atributy a pravidla používání	167
Projekt 5.1: Jednoduchý nákupní vozík#	167
5.8 Akce <jsp:plugin>	173
Atributy a pravidla používání	173
Mapování apletu do akce <jsp:plugin>	175
Projekt 5.2: Vkládání nabídky pomocí apletu#	176
5.9 Závěrečný test	178

Cíle kapitoly:

-  pochopit JSP akce
-  seznámit se standardními akcemi
-  naučit se používat standardní akce efektivně

6. Knihovny značek

6.1 Proč definovat vlastní značky?	180
6.2 Elementy knihovny značek	184
Formáty knihovny značek	184
Direktiva taglib	185
Popisovač knihovny značek	185
Manipulátory značek	189
Výchozí vlastnosti manipulátoru značky	191
Atributy značky	191
Rozhraní Tag	194
Rozhraní BodyTag	196
6.3 Vývojové fáze manipulátoru značky	199
6.4 Definování skriptovacích proměnných	202
Třída TagExtraInfo	202
Definování skriptovacích proměnných	205
Doporučená konvence pro definování implicitních objektů	207
Projekt 6.1: Definování implicitních objektů#	207
6.5 Výměna dat mezi akcemi a knihovnamí značek	211
Názvy objektů a objekt PageContext	211
Běhový zásobník	212
Projekt 6.2: Šablona pomocí knihovny značek#	214
6.6 Závěrečný test	221

Cíle kapitoly:

- ☞ pochopit význam vlastních značek
- ☞ seznámit se s funkcí vlastních značek
- ☞ poznat jak vlastní značky implementovat
- ☞ naučit se vytvořit ze sady vlastních značek knihovnu značek

V kapitole 5 jsme popsali standardní akce, které definuje specifikace JSP. Tato kapitola vysvětluje, jak psát vaše vlastní akce, pomocí kterých můžete množinu standardních akcí a tím i samotný skriptovací jazyk JSP rozšířit. Vlastní značky jsou velmi mocným nástrojem a lze je bez nadsázky považovat za nejdůležitější vlastnost JSP stránek.

7. Architektury JSP

7.1 Vzory v procesu vývoje aplikací.....	225
Idea vzorů.....	225
Využití vzorů.....	226
7.2 Vzory architektury.....	227
Aplikační vrstvy	228
Jednovrstvé architektury	229
Dvojvrstvé architektury	229
Třívrstvé architektury	230
Rozdělení prezentační vrstvy.....	231
Architektury webových aplikací.....	233
7.3 Objektově orientované návrhové vzory.....	235
Model-view-kontroler.....	235
Prostředník (Mediator)	237
Příkaz (Command).....	238
Strategie (Strategy)	240
Most (Bridge)	241
7.4 Projekt 7-1: Prohlížeč obrázků#	243
Krok za krokem	244
7.5 Aplikace: Rodinné centrum	250
Bezpečnost.....	251
Řízení toku aplikace (web-flow)	251
Uložení dat	251
7.6 Závěrečný test.....	252
Vytváření tabulek a vkládání záznamů.....	401

Cíle kapitoly:

- ☞ seznámit se softwarovými vzory
- ☞ porozumět základům architektur webových aplikací založených na Javě
- ☞ porozumět roli JSP v těchto architekturách
- ☞ naučit se používat softwarové vzory pro JSP aplikace
- ☞ ukázat si konkrétní aplikaci, kterou pak použijeme v dalších kapitolách 8, 9 a 10

8. JSP a bezpečnost

8.1 Základy bezpečnosti aplikací	254
Autentikace	254
Bezpečnostní domény a „kruhy důvěry“	255
Autorizace	258
Integrita	259
Utajení	260
Dostupnost	260
Audit	261
8.2 Autentikace na WWW	261
LDAP autentikace	262
Autentikace pomocí certifikátů	263
Autentikace pomocí WWW serveru	265
Konfigurujeme autentikaci s Apache	267
8.3 Autentikace pomocí servlet kontejneru	270
Základní autentikace	271
Konfigurace Tomcatu pro základní autentikaci	271
Autentikace pomocí otisku (digest)	273
Autentikace pomocí HTTPS	273
Autentikace založená na formulářích	273
8.4 Bezpečnostní interceptory Tomcatu	276
Projekt 8-1: Vlastní bezpečnostní interceptor pro Tomcat	278
Projekt 8-2: Vylepšujeme naše přihlašovací stránky	281
8.5 Prvky našeho řešení bezpečnosti	284
Přihlašovací a chybové stránky	284
Soubor web.xml	285
Bezpečnostní interceptor CustomRealm	285
Metoda setContextManager	286
Metoda contextInit	286
Metoda contextShutdown	287
Metoda authenticate	287
Metoda authorize	288
Konfigurační soubor server.xml	288
8.6 Závěrečný test	289

Cíle kapitoly:

- ☞ porozumět základům bezpečnosti software
- ☞ seznámit se s různými druhy zabezpečení webových aplikací
- ☞ seznámit se s bezpečnostními mechanismy JSP a servletů
- ☞ přidat bezpečnostní prvky do naší aplikace Rodinné centrum