

آموزش Java Server Faces



به همراه جدیدترین قابلیت‌های JSF 2.0

مؤلفین: دیوید گری - کای هورستمن

مترجم: مهدی قرمزکن



**NAGHOOS
PUBLICATION**

Geary, David M

عنوان و نام پدیدآور : آموزش JavaServer Faces : به همراه جدیدترین قابلیت‌های Java JSF 2.0

مؤلفین دیوید گری، کای هورستمن؛ مترجم مهدی قرمزکن.

مشخصات نشر : تهران: انتشارات ناقوس، ۱۳۹۰.

مشخصات ظاهری : ۶۰۰ ص: مصور.

شابک : ۹۷۸-۹۶۲-۳۷۷-۵۲۲-۳ : بها: ۱۷۵۰۰۰ ریال

وضعیت فهرست‌نویسی: فیبا

یادداشت : عنوان اصلی: Core JavaServer faces, 3rd ed, c2010.

موضوع : صفحه‌های سرور جاوا

موضوع : وب-سایت‌ها-توسعه

موضوع : وب-سایت‌ها-طراحی

شناسه افزوده : هورستمن، کای اس، ۱۹۵۹-م.

شناسه افزوده : Horstmann, Cay S.

شناسه افزوده : قرمزکن، مهدی، ۱۳۶۰-، مترجم

رده بندی کنگره : TK5۱۰۵/۸۸۸۵/ص۶۰۰ ۱۳۹۰

رده بندی دیویی : ۰۰۶/۷۸

شماره کتابشناسی ملی : ۲۶۹۵۳۶



www.naghoospress.ir



NAGHOOS
PUBLICATION

چاپ اول

نام کتاب : آموزش JavaServer Faces

ناشر : انتشارات ناقوس

مؤلفین : دیوید گری، کای هورستمن

مترجم : مهدی قرمزکن

چاپ اول : ۱۳۹۰

تیراژ : ۱۰۰۰ جلد

لیتوگرافی : گلیا گرافیک

چاپ و صحافی : کنج شایگان

حروف‌نگار و صفحه‌آرا : مریم رضائی

قیمت : ۱۷۵۰۰۰ ریال

شابک : ۹۷۸-۹۶۲-۳۷۷-۵۲۲-۳

ISBN : 978-964-377-522-3

کلیه حقوق برای سرمایه‌گذار محفوظ است. تکثیر، تلافی یا تسعیتی از این اثر به‌صورت حروفچینی یا چاپ مجدد، چاپ افست، پلی‌کپی، فتوکپی و انواع دیگر چاپ ممنوع است و پیگرد قانونی دارد.

مراکز پخش:

۱- انتشارات ناقوس: میدان انقلاب، خیابان کارگر جنوبی، خیابان روانمهر، کوچه دولتشاهی، پلاک ۲

تلفن و فاکس: ۶۶۲۰۸۵۲۰ - ۶۶۲۰۱۷۹۸ - ۶۶۲۱۷۰۷۲ - ۶۶۳۶۶۷۹۳

۲- کتابفروشی الیاس: خیابان انقلاب، نبش فروردین

فهرست مطالب

۱۳	پیشگفتار مؤلفین
۱۷	پیشگفتار مترجم
۱۹	فصل ۱ / JavaServer Face
۱۹	چرا باید Java Server Faces را انتخاب کنیم؟
۲۰	بررسی یک مثال ساده
۲۳	معرفی قسمت‌های مختلف برنامه
۲۴	ساختار دایرکتوری‌ها
۲۵	ایجاد یک برنامه JSF
۲۸	روش اجرای برنامه JSF
۳۰	محیط برنامه‌نویسی JSF
۳۲	تجزیه و تحلیل برنامه ایجاد شده
۳۳	کلاس‌های Bean
۳۴	صفحات JSF
۳۶	فایل‌های تنظیماتی (Configuration Files)
۳۹	نگاهی اجمالی به Ajax JSF 2.0
۴۱	سرویس‌های مختلف فریم‌ورک JSF
۴۴	نحوه عملکرد JSF Implementation
۴۴	نمایش صفحات (Rendering Page)
۴۶	پردازش درخواست ارسال شده
۴۶	طول عمر صفحه JSF (JSF Life Cycle)

نتیجه‌گیری از فصل

۴۸

۴۹

فصل ۲/ کلاس‌های Managed Bean

۴۹

مفاهیم کلاس Bean

۵۲

خاصیت‌های کلاس Bean (Bean Properties)

۵۳

دستورات Value Expression

۵۴

تعریف Backing Bean

۵۵

کلاس‌های CDI Beans

۵۶

فایل‌های Message Bundle

۵۸

تعریف پارامتر در بیتنام‌ها

۵۹

تعریف بیان مجاوره‌ای برنامه (Application Local)

۶۱

ایجاد یک برنامه ساده

۶۷

ناحیه‌های دسترسی کلاس Bean (Bean Scope)

۶۷

ناحیه دسترسی Session (Session Scope)

۶۹

ناحیه دسترسی Request (Request Scope)

۶۹

ناحیه دسترسی Application (Application Scope)

۷۰

ناحیه دسترسی Conversation (Conversation Scope) CDI

۷۱

ناحیه دسترسی View (View Scope) JSF 2.0

۷۱

ناحیه دسترسی Custom (Custom Scope) JSF 2.0

۷۲

پیکربندی کلاس Bean

۷۲

فراخوانی کلاس‌های CDI Bean

۷۳

فراخوانی کلاس‌های Managed Bean JSF 2.0

۷۴

Annotation‌های مربوط به مدیریت طول عمر کلاس Bean

۷۴

تنظیم کلاس Managed Bean با استفاده از دستورات XML

۷۹

قواعد و دستورات عمل‌های Expression Language (EL)

۷۹

معرفی Rvalue و Lvalue

۸۰

استفاده از براکت‌ها

۸۱

دستورات EL برای استفاده از آبجکت‌های List و Map

۸۲

فراخوانی متدها و توابع در دستورات EL JSF 2.0

۸۳

آبجکت‌های رزرو شده در EL

۸۵

دستورات Expression ترکیبی

۸۵

استفاده از متدها در دستورات EL (Method Expression)

۸۶

ارسال پارامتر در دستورات Method Expression JSF 2.0

۸۷

نتیجه‌گیری از فصل

۸۹

فصل ۳/ برقراری ارتباط بین صفحات

۸۹

ایجاد ارتباط ایستا بین صفحات (Static Navigation)

- ۹۰ ایجاد ارتباط پویا بین صفحات (Dynamic Navigation)
- ۹۱ ارتباط Outcome به صفحه وب
- ۹۳ برنامه JavaQuiz
- ۱۰۲ جابه‌جایی بین صفحات یا Redirection
- ۱۰۳ استفاده از آبجکت Flash در هنگام Redirection **JSF 2.0**
- ۱۰۴ تعریف آدرس‌های Bookmarkable و RESTful Navigation **JSF 2.0**
- ۱۰۶ پارامترهای View
- ۱۰۶ ایجاد درخواست‌های GET
- ۱۰۷ ارسال پارامتر در درخواست‌های GET
- ۱۰۸ اضافه نمودن لینک‌های Bookmarkable در برنامه Quiz
- ۱۱۲ تعریف navigation-rule‌های پیشرفته
- ۱۱۳ استفاده از کاراکترهای Wildcard
- ۱۱۴ استفاده از تگ from-action
- ۱۱۴ تعریف Navigation-case‌های شرطی **JSF 2.0**
- ۱۱۴ فراخوانی صفحات به صورت پیمایشی **JSF 2.0**
- ۱۱۵ نتیجه‌گیری از فصل

فصل ۴ / کامپوننت‌های استاندارد JSF

- ۱۱۸ مروری بر تگ‌های کتابخانه JSF Core
- ۱۲۰ تگ‌های ایجادکننده Attribute, Parameter و Facet
- ۱۲۱ مروری بر کامپوننت‌های کتابخانه JSF HTML
- ۱۲۳ خاصیت‌های عمومی کامپوننت‌های کتابخانه JSF HTML
- ۱۳۱ کامپوننت‌های Panel
- ۱۳۴ تگ‌های form, body, head و
- ۱۳۵ استفاده از جاوا اسکریپت برای دسترسی به عناصر فرم
- ۱۳۹ کامپوننت‌های Text Field و Text Area
- ۱۴۲ کامپوننت Hidden Field
- ۱۴۳ استفاده از Text Field و Text Area
- ۱۴۶ کامپوننت‌های نمایش متن و تصویر
- ۱۴۹ کامپوننت‌های Button و Link
- ۱۵۱ کامپوننت‌های Button
- ۱۵۵ کامپوننت‌های Link
- ۱۵۹ تگ‌های انتخابی (Selection Tags)
- ۱۶۲ کامپوننت‌های Checkbox و Radiobutton
- ۱۶۵ کامپوننت‌های Menu و Listbox
- ۱۶۷ تعریف آیتم‌ها (Items)
- ۱۸۴ کامپوننت‌های Message

نتیجه‌گیری از فصل

۱۹۰

۱۹۱

فصل ۵ / استفاده از Facelet در JSF 2.0

۱۹۱

تگ‌های کتابخانه Facelet

۱۹۳

ایجاد Template با استفاده از تگ‌های Facelet

۱۹۴

مراجعه صفحات براساس یک Template عمومی

۱۹۸

سازماندهی صفحات

۲۰۳

استفاده از تگ Decorate

۲۰۵

ارسال پارامتر به Template

۲۰۵

ایجاد تگ‌های سفارشی

۲۰۸

تگ‌های Component و Fragment

۲۰۸

سایر تگ‌های Facelets

۲۰۹

تگ <ui:debug>

۲۱۰

تگ <ui:remove>

۲۱۱

مدیریت فضاها و خالی کردن صفحات

۲۱۲

نتیجه‌گیری از فصل

۲۱۳

فصل ۶ / کامپوننت DataTable

۲۱۴

تگ ایجادکننده Data Table - h:dataTable

۲۱۵

ایجاد یک جدول ساده

۲۱۷

خاصیت‌های تگ h:dataTable

۲۱۹

خاصیت‌های تگ h:column

۲۱۹

تعریف Header, Footer, و Caption

۲۲۲

استایل‌ها

۲۲۳

تعریف استایل برای ستون

۲۲۴

تعریف استایل برای ردیف

۲۲۴

استفاده از تگ ui:repeat در JSF 2.0

۲۲۶

استفاده از کامپوننت‌های استاندارد JSF در h:dataTable

۲۲۹

ویرایش اطلاعات در h:dataTable

۲۲۹

ویرایش سلول‌های جدول

۲۳۲

حذف ردیف در h:dataTable در JSF 2.0

۲۳۵

نمایش رکوردها از پایگاه داده

۲۳۹

کلاس‌های DataModel

۲۳۹

نمایش شماره ردیف

۲۴۰

دسترسی به ردیف انتخاب شده

۲۴۱

مرتب‌سازی و فیلتر نمودن رکوردها

۲۴۸

تکنیک‌های برای ایجاد اسکرول در h:dataTable

۲۴۸	ایجاد اسکرول با استفاده از Scrollbar
۲۴۹	ایجاد اسکرول با استفاده از صفحه‌بندی جدول
۲۵۰	نتیجه‌گیری از فصل

فصل ۷ / اعتبارسنجی (Validation) و تبدیل داده‌ها (Conversion) ۲۵۱

۲۵۲	مروری بر روش اجرای Conversion و Validation
۲۵۳	استفاده از تگ‌های استاندارد Converter
۲۵۳	تبدیل اعداد و تاریخ
۲۵۸	مدیریت پیغام‌های خطا در Converterها
۲۶۴	یک مثال کامل در مورد استفاده از Converterها
۲۶۷	استفاده از تگ‌های استاندارد Validator
۲۶۷	اعتبارسنجی طول یک عبارت رشته‌ای و تعیین محدوده‌ی اعداد
۲۶۸	استفاده از تگ <code>validateRequired</code>
۲۶۹	مدیریت پیغام‌های خطا در Validatorها
۲۷۱	حذف و نادیده گرفتن فاز Validation
۲۷۱	یک مثال کامل در مورد استفاده از Validatorها
۲۷۴	اعتبارسنجی از طریق فیلدهای کلاس Bean JSF 2.0
۲۷۹	ایجاد Validator و Converter مطابق با نظر برنامه‌نویس
۲۸۰	پیاده‌سازی کلاس Converter مطابق با نظر برنامه‌نویس
۲۸۲	تخصیص Converterهای سفارشی ایجاد شده به یک تگ JSF 2.0
۲۸۴	مدیریت پیغام‌های خطا در Converterهای سفارشی
۲۸۵	نمایش پیغام‌های خطا از فایل Resource Bundle
۲۸۹	یک مثال کامل در مورد نحوه‌ی ایجاد کلاس‌های Converter
۲۹۲	تعریف Attribute برای Converterهای ایجاد شده
۲۹۲	پیاده‌سازی کلاس‌های Validator مطابق با نظر برنامه‌نویس
۲۹۴	تخصیص Validatorهای سفارشی ایجاد شده به یک تگ
۲۹۷	اعتبارسنجی با استفاده از متدهای کلاس Bean
۲۹۷	اعتبارسنجی بین چندین کامپوننت مرتبط با هم
۲۹۹	پیاده‌سازی تگ‌های Validator و Converter مطابق با نظر برنامه‌نویس
۳۰۴	نتیجه‌گیری از فصل

فصل ۸ / مدیریت رویدادها ۳۰۵

۳۰۶	رویدادها و طول عمر صفحه JSF
۳۰۷	رویدادهای Value Change
۳۱۲	رویدادهای Action
۳۱۷	تگ‌های ایجادکننده رویداد
۳۱۷	تگ <code>f:actionListener</code> و <code>f:valueChangeListener</code>

- ۳۱۹ استفاده از خاصیت immediate در کامپوننت‌های JSF
- ۳۲۰ استفاده از خاصیت immediate در کامپوننت‌های متنی
- ۳۲۲ استفاده از خاصیت immediate در کامپوننت‌های دستوری
- ۳۲۲ ارسال اطلاعات از صفحه وب به سرور و کلاس‌های Backing Bean
- ۳۲۴ ارسال پارامتر به دستورات Method Expression **JSF 2.0**
- ۳۲۵ استفاده از تگ f:param
- ۳۲۵ استفاده از تگ f:attribute
- ۳۲۶ استفاده از تگ f:setPropertyActionListener **JSF 2.0**
- ۳۲۷ رویدادهای Phase
- ۳۲۸ رویدادهای System **JSF 2.0**
- ۳۳۰ اعتبارسنجی چندین کامپوننت با هم
- ۳۳۲ تصمیم‌گیری قبل از نمایش صفحه وب
- ۳۳۷ ایجاد یک مثال کامل
- ۳۴۳ نتیجه‌گیری از فصل

فصل ۹ / کامپوننت‌های ترکیبی (Composite Component) **JSF 2.0**

- ۳۴۶ کتابخانه تگ‌های ایجادکننده Composite Component
- ۳۴۷ استفاده از یک Composite Component
- ۳۵۰ پیاده‌سازی یک Composite Component
- ۳۵۰ تنظیمات Composite Component
- ۳۵۲ تعیین نوع خاصیت‌ها در یک Composite Component
- ۳۵۳ خاصیت‌های الزامی و مقدار پیش‌فرض برای آنها
- ۳۵۴ ارتباط با داده‌های سرور در Composite Component
- ۳۵۶ استفاده از Resource Bundle در کامپوننت‌های ترکیبی
- ۳۵۷ اعتبارسنجی در Composite Component
- ۳۶۰ استفاده از تگ composite:actionSource
- ۳۶۲ تعریف و ایجاد Facet
- ۳۶۴ اضافه نمودن کامپوننت به بدنه کامپوننت‌های ترکیبی در هنگام استفاده
- ۳۶۵ استفاده از جاوا اسکریپت در Composite Component
- ۳۶۹ مدیریت Composite Component در کلاس‌های جاوا
- ۳۷۸ قراردادن کامپوننت ترکیبی در یک فایل JAR
- ۳۷۹ نتیجه‌گیری از فصل

فصل ۱۰ / استفاده از Ajax **JSF 2.0**

- ۳۸۲ تکنولوژی Ajax و فریم‌ورک JSF
- ۳۸۳ مراحل طول عمر صفحات در حالت استفاده از Ajax
- ۳۸۴ مراحل استفاده از Ajax

۲۸۵	تگ fajax
۲۸۸	استفاده از Ajax برای مجموعه‌ای از کامپوننت‌ها
۲۹۰	استفاده از Ajax برای اعتبارسنجی کامپوننت‌ها
۲۹۱	مانیتور کردن درخواست‌های Ajax
۲۹۴	تعریف Namespace برای توابع جاوا اسکریپت
۲۹۵	مدیریت پیغام‌های خطا در درخواست‌های Ajax
۲۹۶	Ajax Response
۲۹۸	کتابخانه جاوا اسکریپت JSF 2.0
۴۰۰	استفاده از پارامترهای تکمیلی در درخواست‌های Ajax
۴۰۲	تعریف صف برای اجرای رویدادهای Ajax
۴۰۳	تعریف فواصل زمانی در اجرای رویدادهای Ajax
۴۰۴	ایجاد وقفه در jsf.ajax.request()
۴۰۴	استفاده از Ajax در Composite Component‌ها
۴۱۰	نتیجه‌گیری از فصل

فصل ۱۱/ ایجاد کامپوننت‌های سفارشی (Custom Components) F11

۴۱۲	پیاده‌سازی کلاس یک کامپوننت
۴۱۶	مرحله Encoding: طراحی واسط کاربری کامپوننت با استفاده از دستورات Markup
۴۱۹	مرحله Decoding: پردازش درخواست‌های ارسال شده
۴۲۵	فایل Tag Library Descriptor JSF 2.0
۴۲۸	استفاده از External Renderer
۴۳۱	ایجاد و پردازش Attribute‌ها JSF 2.0
۴۳۳	تعریف رویداد Value Change برای کامپوننت
۴۳۴	پشتیبانی از دستورات Method Expression
۴۳۵	تشکیل صف برای اجرای رویدادها
۴۳۶	ایجاد یک برنامه کامل
۴۴۳	پردازش دستورات جاوا اسکریپت (Encoding JavaScript)
۴۴۶	استفاده از Child Component و آبجکت‌های Facet
۴۵۰	تعریف و استفاده از SelectItem
۴۵۰	تعریف و استفاده از آبجکت Facet
۴۵۱	استفاده از Hidden Field
۴۵۷	ذخیره و بازیابی وضعیت‌های مختلف در کامپوننت‌های سفارشی
۴۵۹	ذخیره‌ی بخشی از یک وضعیت (State) JSF 2.0
۴۶۳	استفاده از Ajax در کامپوننت‌ها JSF 2.0
۴۶۴	پیاده‌سازی Ajax در کدهای کلاس کامپوننت
۴۶۸	استفاده از تگ fajax در کامپوننت
۴۷۴	نتیجه‌گیری از فصل

فصل ۱۲ / ارتباط با پایگاه داده و سرویس‌های وب

۴۷۵

- ۴۷۵ دسترسی به پایگاه داده با استفاده از کتابخانه JDBC
- ۴۷۵ استفاده از دستورات SQL
- ۴۷۷ مدیریت آبجکت‌های Connection ایجاد شده با پایگاه داده
- ۴۷۸ مدیریت آبجکت Connection
- ۴۸۰ استفاده از آبجکت PreparedStatement
- ۴۸۱ پیاده‌سازی Transaction
- ۴۸۲ استفاده از پایگاه داده Derby
- ۴۸۴ تنظیم DataSource
- ۴۸۴ دسترسی به DataSource‌های ایجاد شده در Application Server
- ۴۸۵ ایجاد یک DataSource برای اتصال به پایگاه داده در سرور GlassFish
- ۴۸۷ ایجاد یک DataSource برای اتصال به پایگاه داده در سرور Tomcat
- ۴۸۸ ایجاد یک مثال کامل از ارتباط با پایگاه داده
- ۴۹۵ استفاده از Java Persistence Architecture (JPA)
- ۴۹۶ مروری بر JPA
- ۴۹۷ استفاده از JPA در برنامه‌های وب
- ۵۰۲ استفاده از Managed Bean و کلاس‌های Stateless Session Bean
- ۵۰۶  Stateful Session Bean کلاس‌های
- ۵۰۸ مدیریت کاربران و نقش‌های کاربری در Application Server
- ۵۲۰ ارسال ایمیل
- ۵۲۶ استفاده از سرویس‌های وب (Web Service)
- ۵۳۲ نتیجه‌گیری از فصل

۵۳۵

فصل ۱۳ / چگونه می‌توان ...

- ۵۳۵ چگونه می‌توان کامپوننت‌های بیشتری برای استفاده در صفحات JSF پیدا نمود؟
- ۵۳۶ چگونه می‌توان از File Upload در برنامه‌های JSF استفاده نمود؟
- ۵۴۴ چگونه می‌توان از image map در صفحه JSF استفاده نمود؟
- ۵۴۶ چگونه می‌توان داده‌های باینری را در صفحات JSF مورد استفاده قرار داد؟
- ۵۵۴ چگونه می‌توان تعداد زیادی رکورد را در یک صفحه JSF نمایش داد؟
- ۵۵۹ چگونه می‌توان یک پنجره Pop-up طراحی نمود؟
- ۵۶۷ چگونه می‌توان قسمتی از صفحه را برای مخفی‌شدن یا نمایش دادن انتخاب نمود؟
- ۵۶۸ چگونه می‌توان صفحات خطای برنامه را مطابق با نیاز خود تغییر داد؟
- ۵۷۳ چگونه می‌توان تگ‌هایی Client-Side برای اعتبارسنجی کامپوننت‌ها طراحی نمود؟
- ۵۷۹ چگونه می‌توان تنظیمات برنامه را به صورت دینامیک تغییر داد؟
- ۵۸۱ چگونه می‌توان دستورات جدیدی به Expression Language اضافه نمود؟
- ۵۸۴ چگونه می‌توان توابع جدیدی را به دستورات Expression Language اضافه نمود؟ **JSF 2.0**
- ۵۸۵ چگونه می‌توان ترافیک بین سرور و مرورگر را مانیتور نمود؟

- ۵۸۷ چگونه می‌توان ایرادات صفحه را شناسایی و برطرف نمود؟
- ۵۸۸ چگونه می‌توان در هنگام ایجاد برنامه‌های JSF از ابزارهای Test استفاده نمود؟
- ۵۹۰ چگونه می‌توان از زبان Scala در پروژه‌های JSF استفاده نمود؟
- ۵۹۲ چگونه می‌توان از زبان Groovy در برنامه‌های JSF استفاده نمود؟
- ۵۹۳ چگونه می‌توان یک پروژه JSF را در محیط Eclipse ایجاد نمود؟
- ۵۹۶ چگونه می‌توان رکوردهای h:dataTable را به فرمت‌های Excel و PDF ذخیره نمود؟
- ۵۹۸ نتیجه‌گیری از فصل

پیشگفتار مؤلفین

هنگامی که اولین بار در سمینار JavaOne در سال ۲۰۰۲ اطلاعاتی در مورد JavaServer Faces (JSF) شنیدیم، بسیار هیجان زده شدیم. هر دوی ما تجربیات زیادی با برنامه‌نویسی سمت کاربر در جاوا داشته‌ایم. دیوید کتاب *Graphic in Java*، و کای کتاب *Core Java* را که هر دو توسط انتشارات SunMicroSystem چاپ شده‌اند، تألیف کرده‌اند. و برنامه‌های وب را با استفاده از تکنولوژی‌های Servlet و JavaServer Page (JSP) که بسیار سخت و پیچیده بودند، طراحی می‌کردیم. در این سمینار طراحان فریم‌ورک JSF قول داده بودند که، واسط کاربری انعطاف‌پذیری ایجاد کرده، و به جای اینکه برنامه‌نویس را درگیر پردازش‌های متعدد درخواست‌های ارسال شده و مدیریت فرم نمایند، با در اختیار قرار دادن کامپوننت‌هایی مانند کامپوننت‌های منی و منوها قابلیت بیشتری در اختیار آنها قرار داده و سادگی طراحی وب را افزایش دهند. در آن زمان هر کدام، قصد داشتیم کتابی را برای ناشر خود تألیف کنیم، به همین دلیل به انتشارات Sun MicroSystem پیشنهاد دادیم که کتابی در مورد تکنولوژی نوظهور JSF تهیه و ارائه نماییم.

در سال ۲۰۰۴، تیم طراحی JSF (که دیوید نیز یکی از اعضای آن بود) اولین نگارش فریم‌ورک JSF، یعنی JSF 1.0 را برای استفاده برنامه‌نویسان و طراحان سیستم‌های وب بیس ارائه کرده بودند. بسیاری از خطاهای این نگارش به سرعت برطرف گردیده، و با عنوان نگارش JSF 1.1 دوباره عرضه گردید. در سال ۲۰۰۶ نسخه جدیدتری از این فریم‌ورک یعنی JSF 1.2 با قابلیت‌های جدیدتر در وب سایت شرکت Sun برای دانلود در دسترس عموم قرار گرفت.

تمامی فایل‌های آموزشی و راهنماهای ارائه شده از شرکت Sun براساس آخرین نگارش این فریم‌ورک تغییر کرد. متأسفانه این نگارش از فریم‌ورک در خیلی از موارد وارد جزئیات نشد. به عنوان مثال در مورد توابع API و طراحی صفحات با استفاده از آنها، ارسال و مدیریت درخواست‌های با روش GET، و مدیریت پیغام‌های خطا وارد جزئیات نشده، و از طرف دیگر برنامه‌نویس برای Trace کردن صفحه و پیدا نمودن خطاها با مشکلات بسیاری روبرو بوده است.

البته JSF با توجه به نوع طراحی (معماری MVC) و انعطاف‌پذیری بالا مورد توجه بسیاری از برنامه‌نویسان قرار گرفت، به همین دلیل کامپوننت‌ها و فریم‌ورک‌های open-source متعددی مانند *Facelets*، *Ajax4jsf*، *Jboss Seam*، *JSF Template*، *Pretty Faces*، *Richfaces*، *ICEFaces*، و غیره براساس آن طراحی گردید، که می‌توانستند الهام‌بخش قابلیت‌های بیشتری در نگارش جدید JSF باشند.

در سال ۲۰۰۹، نگارش جدید این فریم‌ورک یعنی نسخه ۲.۰ آن براساس تجربیات سایر برنامه‌نویسان و فریم‌ورک‌های open-source فوق ایجاد گردد. JSF 2.0 نسبت به نگارش‌های قبل بسیار ساده‌تر و با سایر تکنولوژی‌های JEE یکپارچگی بیشتری دارد، در حقیقت تمام مشخصات فریم‌ورک

JSF 1.0 با روش‌های بهتر و ساده‌تری به JSF 2.0 منتقل گردید. همچنین، در این نگارش تکنولوژی‌های بیشتری مانند Ajax و REST نیز به برنامه‌نویسی صفحات اضافه گردید.

JSF در حال حاضر یکی از بهترین تکنولوژی‌های Server-side برای طراحی صفحات وب بوده، و طراحان آن تقریباً به تمام قول‌هایی که وعده داده بودند، عمل کردند. در این فریم‌ورک می‌توانید صفحات خود را با استفاده از کامپوننت‌های طراحی شده، ایجاد کرده و آنها را بدون ادغام کدهای markup و جاوا به اجکت‌های کلاس‌ها متصل نماییم. مهمترین مزیت JSF قابلیت افزایش و گسترش کامپوننت‌های آن، و تعداد کامپوننت‌های زیادی می‌باشد که برای این فریم‌ورک طراحی شده است.

به علت اینکه JSF یک تکنولوژی بوده و محصولی واحد به حساب نمی‌آید، هیچ اجباری در استفاده از یک مجموعه خاص از آن نمی‌باشد. می‌توانید از کامپوننت‌ها، ابزارها، و پیاده‌سازهای متعددی که توسط شرکت‌های مختلف تهیه شده است، در برنامه‌های خود استفاده نمایید. فریم‌ورک JSF 2.0 بسیار فریم‌ورک جالبی بوده که کار با آن برای ما بسیار لذت‌بخش و هیجان‌انگیز می‌باشد. امیدواریم که بتوانیم تجربیات و دانش خود را در رابطه با این فریم‌ورک در اختیار شما گذاشته، و بتوانید برنامه‌های وب انعطاف‌پذیرتری طراحی نمایید.

درباره این کتاب

این کتاب برای برنامه‌نویسانی مفید می‌باشد که هدف اصلی آنها، تمرکز بیشتر بر روی طراحی واسط کاربری و لایه Business Logic برنامه می‌باشد. فریم‌ورک JSF براساس servlet های جاوا پیاده‌سازی و ایجاد شده است، اما از نظر برنامه‌نویسان و طراحان این پیاده‌سازی اصلاً مشهود نبوده، زیرا که فقط با کامپوننت‌ها و فرم‌های وب در ارتباط می‌باشند. از طرف دیگر این فریم‌ورک با قابلیت‌هایی که دارد، از بسیاری از فریم‌ورک‌های موجود مانند JSP, servlet و Struts توانمندتر بوده و قابل قیاس با آنها نمی‌باشد.

در نیمه‌ی اول این کتاب، تا شروع فصل هفتم، بر روی تگ‌های JSF و روش‌های استفاده از آنها بحث شده است. این تگ‌ها از نظر عملکرد شبیه تگ‌های HTML بوده و از آنها برای ایجاد قسمت‌های اصلی صفحه وب که در ارتباط با کاربر می‌باشد، استفاده می‌گردد. هر فردی که اطلاعات پایه‌ای در مورد تگ‌های HTML (طراحان وب سایت‌ها) و برنامه‌نویسی جاوا داشته باشد، می‌تواند از این تگ‌ها برای ایجاد برنامه‌های وب استفاده نماید.

در نیمه‌ی اول این کتاب، به‌طور خلاصه با مفاهیم زیر آشنا خواهید شد:

- تنظیم محیط برنامه‌نویسی برای ایجاد برنامه‌های JSF (فصل اول)
- اتصال تگ‌های JSF به Application Logic و کلاس‌های ایجاد شده در آن (فصل دوم)
- ارتباط بین صفحات مختلف برنامه (فصل سوم)
- استفاده از تگ‌های استاندارد JSF (فصل چهارم)
- استفاده از تگ‌های کتابخانه Facelet برای ایجاد یک قالب کلی جهت صفحات (فصل پنجم)
- استفاده از جدول برای نمایش رکوردها (فصل ششم)

• اعتبارسنجی و تبدیل مقادیر کامپوننت‌های متنی (فصل هفتم)

با شروع فصل هشتم، ما برنامه‌نویسی JSF را به صورت جدی آموزش می‌دهیم. در این فصل یاد خواهید گرفت که چگونه قابلیت‌های بیشتری به فریم‌ورک JSF اضافه نموده و کدهای پیشرفته‌تری را در برنامه‌های خود اجرا نمایید. در نیمه‌ی دوم این کتاب، مفاهیم زیر را به شما آموزش خواهیم داد:

• مدیریت رویدادها (فصل هشتم)

• ایجاد کامپوننت‌های ترکیبی - کامپوننت‌هایی با قابلیت استفاده مجدد که براساس یک یا چند کامپوننت استاندارد JSF برای ایجاد رفتار و عملکردی متفاوت طراحی شده‌اند (فصل نهم) **NEW**

• Ajax (فصل دهم) **NEW**

• پیاده‌سازی و طراحی کامپوننت‌های JSF مطابق با نظر برنامه‌نویس (فصل یازدهم)

• اتصال به پایگاه داده و استفاده از منابع خارجی مانند ایمیل و سرویس‌های وب (فصل دوازدهم)

در آخرین فصل این کتاب نیز به سؤالات عمومی که ممکن است برای شما پیش بیاید در قالب سؤالاتی با عنوان "چگونه می‌توانم..." پرداخته‌ایم (فصل سیزدهم). حتی اگر از قبل با فریم‌ورک JSF آشنا باشید، پیشنهاد می‌کنیم سؤالات این فصل و پاسخ آنها را مطالعه کنید، زیرا که مطمئناً با نکات جالبی روبرو خواهید شد. در این فصل شما را با روش‌های مختلف debug کردن صفحات و قابلیت‌هایی که در JSF نادیده گرفته شده‌اند، مانند ایجاد پیجره‌های pop-up، طراحی و استفاده از file upload، صفحه‌بندی جداول برای نمایش تعداد زیادی از رکوردها و غیره آشنا خواهیم کرد.

نرم‌افزارهای موردنیاز

تمام نرم‌افزارهایی که در این کتاب به آن نیاز خواهید داشت، به صورت رایگان در دسترس می‌باشند. می‌توانید از Application Server های مبتنی بر 6 Java EE (مانند *GlassFish* نگارش سوم) و یا اجراکننده‌های servlet (مانند *Tomcat 6*) استفاده نمایید. این نرم‌افزارها مستقل از سیستم‌عامل و بر روی سیستم‌عامل‌های Solaris، Mac OS X، Linux و ویندوز قابل اجرا می‌باشند. هر دو ابزار *Eclipse* و *NetBeans* پشتیبانی کاملی از فریم‌ورک JSF داشته و می‌توانید برنامه‌های خود را توسط آنها، برای اجرا در محیط *GlassFish* و یا *Tomcat* ایجاد و طراحی نمایید.

پشتیبانی از طریق وب سایت

وب‌سایتی که برای این کتاب طراحی شده است: <http://corejsf.com> بوده و شامل بخش‌های زیر می‌باشد:

- تمام مثال‌های این کتاب به همراه سورس کدها آن.
- لیستی از خطاهای پیدا شده در این کتاب و سورس کدها برنامه.
- لیستی از منابعی که فکر کردیم، ممکن است نسبت به نسخه چاپی کارایی بیشتری داشته و برای شما سودمندتر باشند.
- صفحه‌ای برای ارائه پیشنهادات و انتقادات شما از این کتاب.

پیشگفتار مترجم

فریم‌ورک JavaServer Face بدون شک یکی از بهترین فریم‌ورک‌های وب‌بیس برای طراحی صفحات وب به صورت Server-Side بوده که توسط آن می‌توان برنامه‌های مبتنی بر وب بسیار پیشرفته‌ای ایجاد نمود. این فریم‌ورک با توجه به معماری MVC (Model-View-Controller) انعطاف بالاتری نسبت به فریم‌ورک‌های PHP، JSP داشته، زیرا که در این فریم‌ورک بین لایه‌های برنامه مختلف تمایز قائل شده و می‌توان طراحی هر لایه را به تنهایی جداگانه محول نمود.

این کتاب توسط دو تن از بهترین برنامه‌نویسانی که در شرکت‌های Sun و Oracle مدیریت سیستم‌های مختلفی را برعهده داشته‌اند تألیف و در آن سعی شده است، به تمامی نکات لازم که در هنگام برنامه‌نویسی JSF ممکن است به آن نیاز داشته باشید، اشاره شود. اگر از قبل با برنامه‌نویسی JSF آشنا می‌باشید، بهتر است یکبار این کتاب را مطالعه نمایید، زیرا که با مطالعه‌ی این کتاب مطمئناً با نکات جدیدتری آشنا خواهید شد. اگر با محدودی با طراحی صفحات وب و تگ‌های HTML آشنا می‌باشید و قصد دارید که برنامه‌نویسی جاوا را با تکنولوژی JSF یاد بگیرید، بدون شک بهترین گزینه شما این کتاب خواهد بود.

با توجه به روند رو به رشد جاوا و پشتیبانی شرکت‌هایی مانند Oracle و IBM از آن و همچنین گسترش روز افزون برنامه‌های Open-source در صورت داشتن اطلاعاتی در زمینه برنامه‌نویسی جاوا می‌توانید بازار کار مناسبی در ایران، و یا در کشورهای دیگر به‌دست آورید. یکی از مزایای JSF در مقیاس با فریم‌ورک‌ها و تکنولوژی‌های NET، مستقل از پلتفرم بودن آن می‌باشد. از طرف دیگر طبق تجربیات به‌دست آمده توسط اینجانب و سایر برنامه‌نویسان، برنامه‌های وب بیس JSF از سرعت، انعطاف و قابلیت‌های بیشتری نسبت به برنامه‌های ایجاد شده در ASP.NET برخوردار می‌باشند.

یکی از نقاط قوت این کتاب که ممکن است در نگاه اول از دید خوانندگان موجب زیاد شدن حجم کتاب شده باشد، کدهای نوشته شده در متن کتاب می‌باشد. این کدها و دستورات بسیار کاربردی بوده که می‌توانید در طول مطالعه کتاب و هنگامی که به کامپیوتر دسترسی ندارید از آنها استفاده نمایید. بسیاری از کدهای این کتاب برای اولین بار توسط مؤلفین ایجاد و سپس بر روی اینترنت توزیع شده‌اند. در این کتاب سعی شده است برای آشنایی خوانندگان گرامی با اصطلاحات برنامه‌نویسی JSF از لغات و کلمات انگلیسی استفاده شود و در بعضی موارد نیز معادل فارسی آنها ذکر گردیده است. اگر قصد دارید که به عنوان یک برنامه‌نویس حرفه‌ای مطرح شوید، بهتر است از الان با این اصطلاحات آشنا گردید.

در تمام قسمت‌های مختلف این کتاب سعی شده است که، نکاتی را که از دید مؤلفان کتاب به‌دور مانده در قالب توضیحاتی با مشخصه مترجم (مترجم) بیان شود. همچنین علاوه بر ابزارهای گفته شده در کتاب می‌توانید از ابزارهای زیر نیز، برای ایجاد برنامه‌های JSF استفاده نمایید:

- ابزار برنامه‌نویسی Eclipse IDE For Java EE Developers یا نگارش‌های بالاتر از آن. (با توجه به اینکه NetBeans حافظه‌ی بیشتری در هنگام طراحی برنامه نیاز خواهد داشت، استفاده از آن در پروژه‌های بزرگ پیشنهاد نمی‌گردد).

<http://eclipse.org/downloads/>

- استفاده از پلاگین JBoss Tools که توسط شرکت JBoss برای ابزار Eclipse طراحی شده است. این ابزار شامل آیین‌هایی برای ایجاد کلاس‌های EJB براساس جدول‌های پایگاه داده، ایجاد فایل‌های xhtml، ایجاد یک پروژه jsf با تمام تنظیمات آن و غیره می‌باشد.

<http://www.jboss.org/tools/download>

- در صورت اینکه کامپوننت‌های استاندارد JSF نتوانند نیاز شما را برطرف نمایند، بهتر است از مجموعه کامپوننت‌های Richfaces یا Primefaces استفاده نمایید.

<http://www.jboss.org/richfaces/download>

<http://www.primefaces.org/downloads.html>

- ایجاد و طراحی گزارش‌ها در برنامه‌های وب یکی از مشکلاتی است که اغلب برنامه‌نویسان با آن برخورد می‌کنند. برای رفع این مشکل پیشنهاد می‌شود از مجموعه ابزارهای رایگان Jasper Report استفاده شود. ابزار aReport برای رایگان برای تهیه گزارشات مبتنی بر Jasper Report بوده که توسط آن می‌توانید گزارشات خود را به صورت کاملاً ری‌وزال طراحی نمایید.

<http://sourceforge.net/projects/jasperreports/files/jasperreports/>

- سرور Oracle GlassFish V3.1 برای اجرای برنامه‌های JSF

<http://glassfish.java.net/>

البته Application Server JBoss AS 7.0 نیز دارای تمام قابلیت‌های GlassFish بوده و از سرعت بوت

و اجرای بالاتری نیز برخوردار می‌باشد.

در پایان لازم می‌دانم که بگویم هیچ کاری بدون نقص و عیب نخواهد بود. از شما خواننده گرامی خواهشمندم هرگونه مشکل در متن این کتاب و یا ایراداتی را که ممکن است در طول مطالعه به آن برخورد کنید، و یا هرگونه انتقادات و پیشنهادات را خود به آدرس ایمیل ghermekzon@gmail.com ارسال کنید.