

## Inhoud eindtoets

### **Eindtoets**

Introductie 2

Opgaven 3

Terugkoppeling 9

## Eindtoets

### INTRODUCTIE

Deze eindtoets is bedoeld als voorbereiding op het tentamen van de cursus Webapplicaties: de serverkant en is te beschouwen als proeftentamen. Het is belangrijk dat u de eindtoets pas probeert te maken op het moment dat u denkt klaar te zijn met de tentamenvoorbereiding. Hebt u over dat laatste nog twijfels, bekijk dan nog eens de leerdoelen en bestudeer de samenvattingen van de hoofdstukken in het tekstboek om te ontdekken welke onderdelen u nog onvoldoende beheerst.

Toegestane hulpmiddelen

Bij het tentamen mag u *geen* cursusmateriaal raadplegen. Ook bij het maken van deze eindtoets mag u dus geen cursusmateriaal gebruiken.

Tentamenduur

Een tentamen duurt drie uur. We adviseren u dan ook de eindtoets binnen een aaneengesloten periode van drie uur te maken. We verwachten overigens dat u deze tijd in het algemeen niet nodig zult hebben.

Samenstelling

Het aantal opgaven, de moeilijkheidsgraad en de verdeling over de leerstof komen overeen met het tentamen. Het tentamen bestaat uit twintig vragen, soms onderverdeeld in deelvragen, waarop een kort antwoord (hoogstens enkele regels) volstaat. U hoeft op het tentamen geen lange stukken programmacode te schrijven (soms wel enkele regels). Soms wordt u wel gevraagd een codefragment te lezen en het effect ervan te beschrijven of uit gegeven onderdelen code samen te stellen. U hoeft in ieder geval geen syntaxis uit het hoofd te leren.

Terugkoppeling

De antwoorden op de opgaven staan in de terugkoppeling. We willen echter benadrukken dat u het meest leert als u eerst de opgaven maakt en pas daarna de antwoorden controleert.

Beoordeling

Het aantal punten dat u per opgave kunt behalen, staat bij de opgave vermeld. U kunt in totaal maximaal 100 punten halen. Voor een voldoende voor het tentamen moet u tenminste 55 punten behalen.

#### *Studeeraanwijzingen*

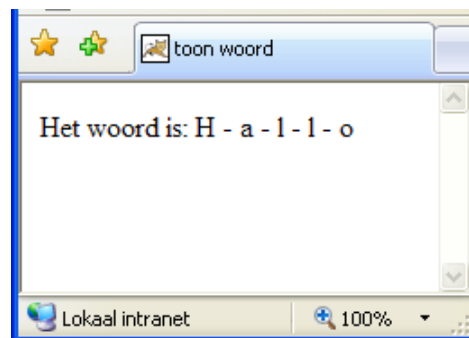
De studielast van deze eindtoets bedraagt circa 4 uur, inclusief het nakijken van de opgaven aan de hand van de terugkoppeling.

### Opgaven

- 4 punten 1 Noem twee belangrijke kenmerken van de communicatie verzorgd door het TCP/IP-protocol.
- 4 punten 2 Welke van de volgende HTTP-berichten bevatten een body?  
- HTTP-GET-request  
- HTTP-POST-request  
- HTTP-response
- 6 punten 3 a Wat zijn de namen van de twee methoden in de klasse HttpServlet die verantwoordelijk zijn voor het afhandelen van een HTTP-request en die in een servlet-klasse gedefinieerd worden?  
b Wat is de functie van elk van deze methoden?  
c Deze twee methoden hebben elk twee parameters. Wat representeren deze parameters?
- 6 punten 4 a Waarvoor wordt een JSP-expressie gebruikt?  
b Waarvoor wordt een JSP-directive gebruikt?
- 5 punten 5 Een HTML-pagina bevat het volgende formulier:

```
<form action="toonWoord.jsp" method="get">  
Geef een woord: <br>  
<input type="text" name="woord">  
<input type="submit" value="Verzenden">  
</form>
```

Als de gebruiker de invoer Hallo geeft en daarna op de verzendknop klikt, dan verschijnt de volgende JSP-pagina waarop tussen twee opvolgende letters van het invoerwoord een streepje staat:



De onvolledige code van de JSP-pagina toonWoord.jsp die hiervoor verantwoordelijk is luidt:

```
<p>Het woord is:  
...  
</p>
```

Op de plaats van de drie puntjes moet JSP-code worden opgenomen om het invoerwoord te tonen met streepjes als scheidingstekens.

Geef het codefragment dat op de JSP-pagina verantwoordelijk is voor het tonen van de tekstregel die de gebruiker te zien krijgt in de browser.

*Aanwijzingen:*

- Scriptlets worden op een JSP-pagina tussen de tags `<% en %>` geplaatst en expressies tussen de tags `<%= en %>`.
- Maak voor het opstellen van de JSP-code gebruik van de volgende Java-code:

```
String woord = request.getParameter("woord");  
for (int i = 0; i < woord.length(); i++) {  
    System.out.print(woord.charAt(i));  
    if ((i < woord.length() - 1) {  
        System.out.print("-");  
    }  
}
```

5 punten

6 Een servlet wordt door de volgende URL aangeroepen:

`http://www.ou.nl:8080/was/tekstServlet?tekst=alfabet`

Beschrijf hoe de webserver de juiste servlet kan aanroepen en de waarde van de parameter door kan geven aan de servlet.

4 punten

7 Welke van de drie volgende beweringen is waar?

- a Een servlet kan alleen via een GET-request worden aangeroepen.
- b Een servlet kan alleen via een POST-request worden aangeroepen.
- c Een servlet kan zowel door een GET-request als door een een POST-request worden aangeroepen.

5 punten

8 Beschrijf kort de structuur van een webapplicatie met de model-1 architectuur.

5 punten

9 Wat zijn de twee belangrijkste overwegingen bij het ontwerpen van de controller van een webapplicatie volgens het MVC-patroon?

5 punten

10 Noem bij sessiebeheer drie manieren waarop een sessie kan eindigen.

5 punten

11 Een bedrijf voor technische apparaten beschikt over een database met informatie over haar producten. Per product wordt het productnummer, de productomschrijving en productinformatie bijgehouden. Het bedrijf wil een webapplicatie laten ontwikkelen opdat medewerkers via het internet de informatie over producten kunnen raadplegen. Voert een medewerker een productnummer in, dan komt een pagina in de browser met alle informatie over het product uit de database. Bij een verkeerd ingevoerd productnummer krijgt de gebruiker een melding. De eerste versie van de webapplicatie vraagt nog geen inloggegevens en is dus voor iedereen toegankelijk. Uitbreidingen volgen later. De webapplicatie moet volgens MVC ontwikkeld worden met gebruik van servlets en JSP.

Beschrijf kort de inhoud van het model, de view en de controller. Geef daarbij aan hoeveel onderdelen elke laag bevat en geef een beknopt omschrijving van elk onderdeel.

- 4 punten 12 a Een singleton-klasse bevat altijd een methode getInstance. Wat is de functie van deze methode?  
b Wat is de toegangspecificatie van deze methode?

- 6 punten 13 Gegeven is de volgende Java-klasse:

```
public class Figuur
{
    private int lengte = 0;
    private int breedte = 0;

    public int getLengte() {
        return lengte;
    }

    public int getBreedte() {
        return breedte;
    }

    public int getOppervlak() {
        return lengte * breedte;
    }

    public void setLengte(int l) {
        lengte = l;
    }

    public void setBreedte(int b) {
        breedte = b;
    }
}
```

- a Is deze klasse een JavaBean? Motiveer kort uw antwoord.  
b Noem de properties van de klasse Figuur.

- 5 punten 14 Gegeven is de volgende referentiekaart met impliciete EL-objecten en hun scope:

Scope	Impliciet EL-object
page	pageScope
request	requestScope
session	sessionScope
application	applicationScope

Welke van de drie volgende EL-expressies zijn equivalent? Motiveer uw antwoord.

- a `${pageScope.cursus.titel}`  
b `${cursus.titel}`  
c `${pageScope.cursus["titel"]}`

- 5 punten 15 Gegeven is de volgende JSP-code:

```

<%@ page import="model.Cursus" %>
<%
    Cursus cursus = (Cursus) session.getAttribute("cursus");
    String lijstUrl = "lijst?cursusCode=" + cursus.getCode();
%>
<a href="<%=response.encodeURL(lijstUrl)%>"Voeg toe</a>

```

Toon code met dezelfde functionaliteit die gebruikmaakt van JSTL en EL.

*Aanwijzing:* Gebruik de volgende referentiekaart:

Element	Syntaxis
EL-expressie	<code>\${...}</code>
JSTL url-tag	<pre> &lt;c:url value='...' /&gt; of &lt;c:url value='...'&gt; ... &lt;/c:url&gt; </pre>
JSTL param-tag	<code>&lt;c:param name='...' value='...' /&gt;</code>
JSTL forEach-tag	<pre> &lt;c:forEach var="..." items="..."&gt;     ... &lt;/c:forEach&gt; </pre> <p>Binnen de tag kunnen tevens de volgende attributen worden gebruikt: <i>begin</i>, <i>end</i>, <i>step</i>, <i>varStatus</i>.</p>
JSTL forTokens-tag	<pre> &lt;c:forTokens var="..." items="..." delims="..."&gt;     ... &lt;/c:forTokens&gt; </pre>
JSTL if-tag	<pre> &lt;c:if test="..."&gt;     ... &lt;/c:if&gt; </pre>
JSTL choose-tag	<pre> &lt;c:choose&gt;     &lt;c:when test="..."&gt;         ...     &lt;/c:when&gt;     ...     &lt;c:otherwise&gt;         ...     &lt;/c:otherwise&gt; &lt;/c:choose&gt; </pre>
JSTL import-tag	<code>&lt;c:import url="..." /&gt;</code>
JSTL out-tag	<pre> &lt;c:out value="..." /&gt; of &lt;c:out value="..." &gt;     ... &lt;/c:out&gt; </pre> <p>Binnen de tag kan tevens het attribuut <i>default</i> worden gebruikt.</p>

4 punten

16 Gegeven is het volgende JSP-fragment:

```
<c:out value="{weerbericht}">  
  Vandaag regent het.  
</c:out>
```

Wat wordt in de browser getoond?

- 6 punten      17    a Geef een omschrijving van de benadering "programmatic approach" voor de ontwikkeling van webapplicaties.  
                      b Geef één techniek die tot deze categorie behoort.
- 5 punten      18    Noem drie criteria die gehanteerd kunnen worden bij de evaluatie van een techniek voor het ontwikkelen van webapplicaties.
- 6 punten      19    a Geef een omschrijving van de functie van de Struts-klasse Action.  
                      b Behoort in een Struts-applicatie een instantie van deze klasse (of van een subklasse daarvan) tot het model, de view of de controller?
- 5 punten      20    Geef een omschrijving van het begrip scaffolding in verband met Ruby on Rails.

## TERUGKOPPELING

### Uitwerking van de opgaven

- 1 Communicatie verzorgd door het TCP/IP-protocol is
  - betrouwbaar
  - connection-oriented
  - gebaseerd op volledige berichten (in tegenstelling tot pakketjes)
  - van eindpunt tot eindpunt (in tegenstelling tot van router tot router)
- 2 HTTP-POST-request en HTTP-response bevatten een body. Merk op dat deze body leeg kan zijn. Na de lege regel volgt dan niets. Een HTTP-GET-request bevat nooit een body.
- 3
  - a De methoden heten doGet en doPost.
  - b De methode doGet, handelt een GET-request af en doPost, een POST-request.
  - b Eén parameter (van type HttpServletRequest) representeert een HTTP-request en de andere parameter (van type HttpServletResponse) representeert een HTTP-response.
- 4
  - a Een JSP-expressie wordt gebruikt om in de HTML-code van een JSP-pagina de string-waarde van een expressie te tonen.
  - b Een JSP-directive wordt gebruikt om condities op te geven die geldig zijn in de hele JSP-pagina.

- 5 De code luidt:

```
<p>Het woord is:  
<%  
    String woord = request.getParameter("woord");  
    for (int i = 0; i < woord.length(); i++) {  
%>  
<%= woord.charAt(i) %>  
<%  
    if (i < woord.length() - 1) {  
%>  
-  
<%  
    }  
%>  
<%  
    }  
%>  
</p>
```

NB: de laatste twee accolades kunnen in een scriptlet worden samengebracht.

- 6 Het bestand web.xml beschrijft de configuratie van de webserver. Daarin is informatie opgenomen over de servlets en de servlet-mapping. De informatie over een servlet vertelt wat de servletnaam is en welke servlet-klasse daarbij hoort. De servlet-mapping vertelt welke URL en welke servletnaam bij elkaar horen.  
(Methode service van de (instantie van de) servlet wordt aangeroepen in



een nieuwe thread.)

De waarde van de parameter tekst wordt doorgegeven via een instantie van type `HttpRequest`.

NB: Dit antwoord is een voorbeeld van een gedetailleerd antwoord. We geven hierna een antwoord dat minder gedetailleerd is maar dat ook wordt goedgekeurd. In de vraagstelling zijn details namelijk niet expliciet gevraagd.

De webserver gebruikt het bestand `web.xml` om met behulp van een url-pattern de gegeven url af te beelden op een servlet. De parameter wordt doorgegeven in de parameter `request` van de methode `doGet` van deze servlet.

- 7 De derde bewering is waar: een servlet kan zowel door een GET-request als door een een POST-request worden aangeroepen.
- 8 Volgens het tekstboek MS bestaat een webapplicatie volgens de model-1 architectuur uit JSP-pagina's, Java-classes en dataopslag (facultatief). Er wordt geen gebruik gemaakt van servlets. Meer algemeen is het belangrijkste kenmerk van de model-1 architectuur dat het verwerken van de request en het genereren van de response in één element plaatsvinden. Een applicatie met alleen een servlet (zonder JSP's voor de view) is ook model 1. Maar dit antwoord wordt niet van u verwacht.
- 9 Eén overweging is het aantal servlets en hun taak. Men kan kiezen om voor elke menuoptie een aparte servlet aan te roepen of om telkens één servlet aan te roepen die dan zelf beslist welke Java-klasse (of servlet) de afhandeling van het verzoek moet uitvoeren. De andere overweging is welke requests via een GET-methode en welke requests via een POST-methode moeten worden afgehandeld.
- 10 Een sessie kan beëindigd worden bij het afsluiten van de browser. Een sessie kan ook beëindigd worden bij het optreden van een timeout, dat wil zeggen als er een bepaalde tijd geen activiteit heeft plaats gevonden, of wanneer de servlet een sessie expliciet beëindigt door de aanroep van methode `invalidate`.
- 11 De controller bevat één servlet die de inkomende verzoeken afhandelt. De servlet bepaalt welke view naar de client wordt verstuurd: een view met informatie bij een correct productnummer of een view met een melding bij een foutieve invoer. De view bevat tot drie JSP-pagina's. Eén pagina biedt de mogelijkheid om een productnummer in te voeren, één pagina toont de gevraagde informatie en één pagina toont een foutmelding. De pagina's kunnen eventueel gecombineerd worden. Het model bevat een `JavaBean` die een afbeelding is van een rij in de database en een klasse voor de toegang tot de database.
- 12
  - a De methode levert de unieke instantie van de klasse op.
  - b De methode heeft als toegangsspecificatie `public static`. (De methode kan ook `synchronized` zijn)

- 13 a Ja, de klasse is een JavaBean. Deze klasse heeft een impliciete parameterloze constructor. De attributen hebben geen toegang public en voor de attributen bestaan get- en set-methoden.  
b De klasse Figuur heeft drie properties: lengte, breedte en oppervlak.
- 14 Alleen de expressies a en c zijn equivalent. Ze leveren de waarde op van de property titel van het attribuut cursus dat opgeslagen is in de page scope.  
Als er in de page scope geen attribuut cursus is opgeslagen leveren de eerste en de derde expressie de lege string op.  
De tweede expressie is alleen maar equivalent met de eerste en de derde expressie als de page scope een attribuut cursus bevat. Anders levert de expressie de titel op van een cursus opgeslagen in een andere (ruimere) scope.
- 15 De code luidt:
- ```
<a href="
  <c:url value='/lijst'>
    <c:param name='cursusCode' value='${cursus.code}' />
  </c:url>">
Voeg toe</a>
```
- of
- ```
<a href="
  <c:url value='/lijst?cursusCode=${cursus.code}' />">
Voeg toe</a>
```
- 16 Als de waarde van het attribuut weerbericht niet null is, wordt deze waarde getoond. Is de waarde null, dan wordt de zin "Vandaag regent het." getoond.
- 17 a In deze benadering bestaat de bron van de webpagina voornamelijk uit programmeercode waarin HTML-code is opgenomen.  
b Servlets (en CGI-scripts) werken volgens deze benadering.
- 18 De volgende criteria kunnen gehanteerd worden:
- Onder welke benadering (programmeerbenadering, template benadering, hybride benadering of framework-benadering) kan de techniek worden ingedeeld?
  - Is de techniek code-centric of page-centric?
  - Is de techniek open source of niet?
  - Welke infrastructuur en software is er nodig?
  - Wat is ermee mogelijk?
  - Hoe complex is de techniek om aan te leren en te gebruiken?
  - Is bij frameworks de techniek overeenkomstig MVC?
- 19 a De klasse Action representeert een afhandeling van een inkomende request. Bij de afhandeling worden eerst klassen van het model aangeroepen en daarna de nodige informatie aan de view doorgegeven. Een Struts-applicatie zal altijd subklassen van Action gebruiken. De controller servlet roept, indien van toepassing, voor een gegeven action de methode execute van de Action-klasse aan.  
b Instanties van subklassen van Action behoren tot de controller.

- 20 Scaffolding is een methode die op basis van een omschrijving van het gebruik van een database een basisapplicatie met mappenstructuur en programmabestanden genereert voor de toegang tot de database.