

Webアプリケーション開発演習A

セッション

情報システム専攻科2年

リクエストスコープ セッションスコープを理解する

- 様々な画面で**共通で使いたい値**は
リクエストスコープでは実現不可能
→セッションスコープでなら実現可能

• セッションを使用するのは3STEP

1. セッションの取得 `HttpSession session = request.getSession(true);`
2. セッションへのセット `session.setAttribute("loginName", "nishino");`
3. セッションからの取得 `(String)session.getAttribute("loginName");`



今日の目標！

**セッション・リクエストを
用いた演習！**

A decorative blue header at the top of the slide, featuring various white geometric shapes like circles, squares, and diamonds, some of which are nested or overlapping.

目次

セッションの復習 もっとセッションを知る 演習

A decorative blue header at the top left of the slide, featuring various white geometric shapes like circles, squares, and diamonds, some with internal patterns.

目次

セッションの復習

もっとセッションを知る

演習

セッションの復習

セッションについて、復習をしましょう
すごく重要なので
しっかり聞いておいてください！





ポイント

セッションの復習

リクエストスコープ

・・・複数の画面で共通して値を使いたいときは×

セッションスコープ

・・・複数の画面で共通して値を使いたいとき○



ポイント

セッションの復習

リクエストスコープ

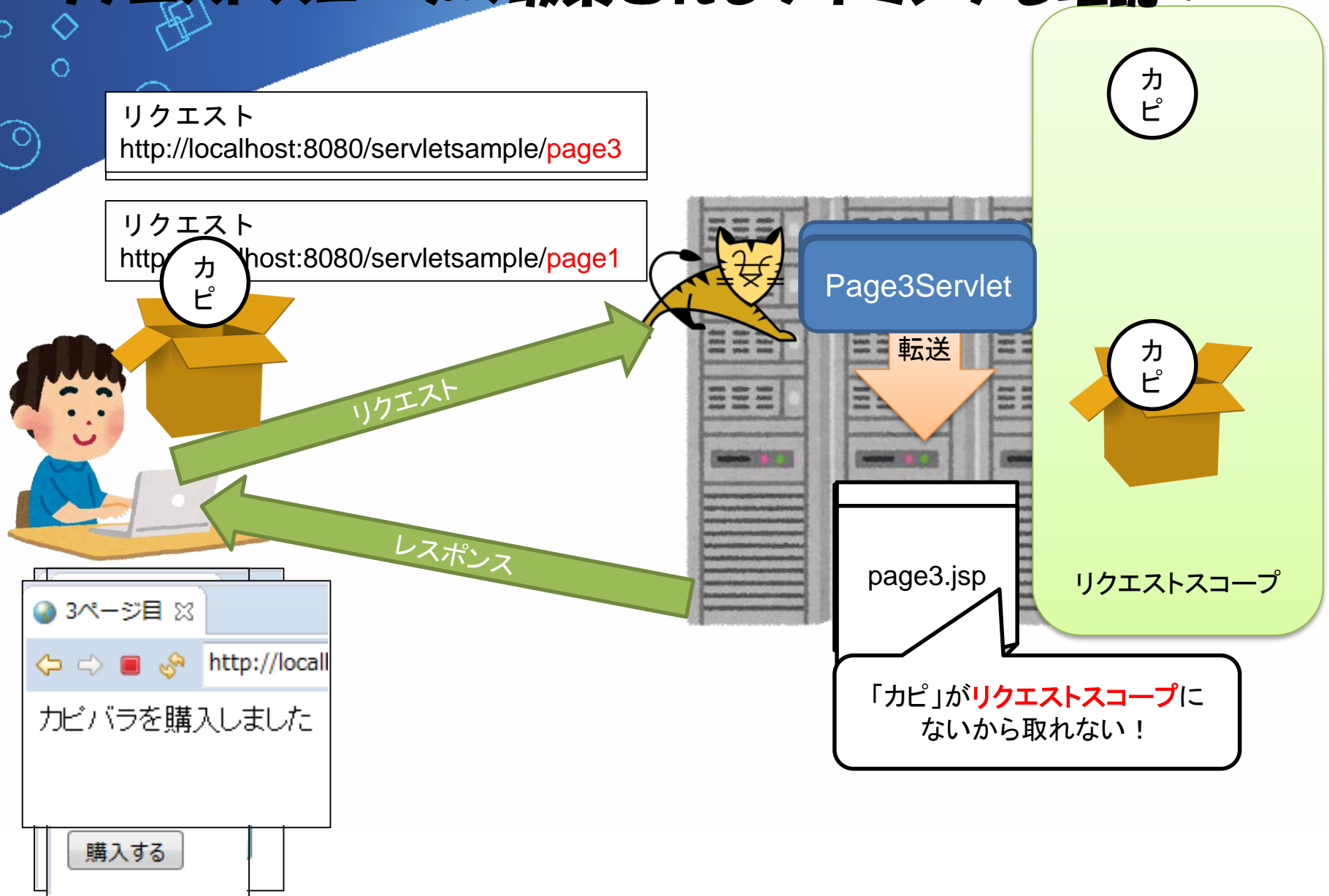
…レスポンスを返すと削除される

セッションスコープ

…レスポンスを返しても削除されない

セッションの復習

リクエストスコープの破棄されるタイミングを確認！



セッションの復習

一方でセッションはずっとあることに注目

リクエスト

<http://localhost:8080/servletsample/page3>

リクエスト

<http://localhost:8080/servletsample/page1>

リクエスト

レスポンス

Page3Servlet

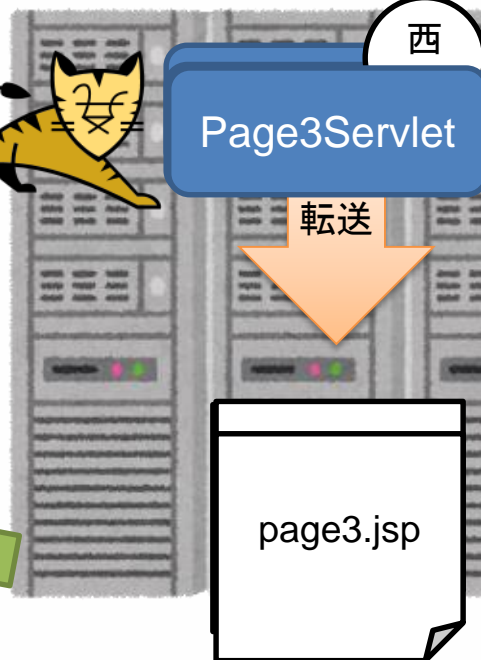
転送

page3.jsp

西

西野

セッションスコープ



セッションの復習

実装方法を振り返ってみよう

セッションに関する実装は3ステップです

STEP1.セッションを取得

STEP2.セッションに値を入れる

STEP3.セッションから値を取得する

ポイント

セッションの復習

STEP1.セッションを取得

//セッションを取得する↓

```
HttpSession session = request.getSession(true);↓
```

セッションは request.getSessionで取得する！！

ポイント

セッションの復習

STEP2.セッションに値を入れる

//セッションに値をセットする↓

```
session.setAttribute("loginName", "nishino");
```

getSessionで取得した
HttpSessionのインスタンス

箱の名前

入れたい値

セッションへは setAttributeでセットする！！

ポイント

セッションの復習

STEP3.セッションから値を取得する

getSessionで取得した
HttpSessionのインスタンス？

箱の名前

```
String loginName = (String)session.getAttribute("loginName");
```

実はJSPでは、書かなくても
裏で勝手に getSessionしてくれています。
なので、getSessionを書かずにいきなり
sessionという名前でインスタンスを使えます。
これを「**暗黙オブジェクト**」といいます。

セッションからはgetAttributeで取得する！

セッションの復習

リクエストとセッションの違いのまとめ

	設定・取得	範囲 (スコープ)	用途
セッション	設定: setAttribute 取得: getAttribute	1度セットすると 複数ページで参照可能	ログインした人の情報や買い物かごの中身など複数のページから参照したい情報を扱いたいとき
リクエスト	設定: setAttribute 取得: getAttribute	1回のリクエストでのみ有効	JSP→サーブレット サーブレット→JSP など1ページの内で情報をやりとりするとき

やってみよう!

演習

**リクエスト/セッションを含め
サーブレットの決まりごとまとめたPDFを作りました。**

**[https://github.com/nishino-naoyuki/2019Web
•Sevletの決まり.pdf](https://github.com/nishino-naoyuki/2019Web-Sevletの決まり.pdf)**



A decorative blue header at the top of the page, featuring various white geometric shapes like circles, squares, and diamonds, some of which are nested or overlapping.

目次

セッションの復習 もっとセッションを知る 演習

もっとセッションを知る

リクエストスコープ

...

セッション

リクエストスコープ必要??
サーブレット→JSPのやりとりは
全部セッションでやればよくな
い??

もっとセッションを知る

**【問】リクエストスコープ必要??
サーブレット→JSPのやいといは
全部セッションでやればよくない??**

【ANSER】

だめに決まとうもん！





ポイント

もっとセッションを知る

理由を教えます。

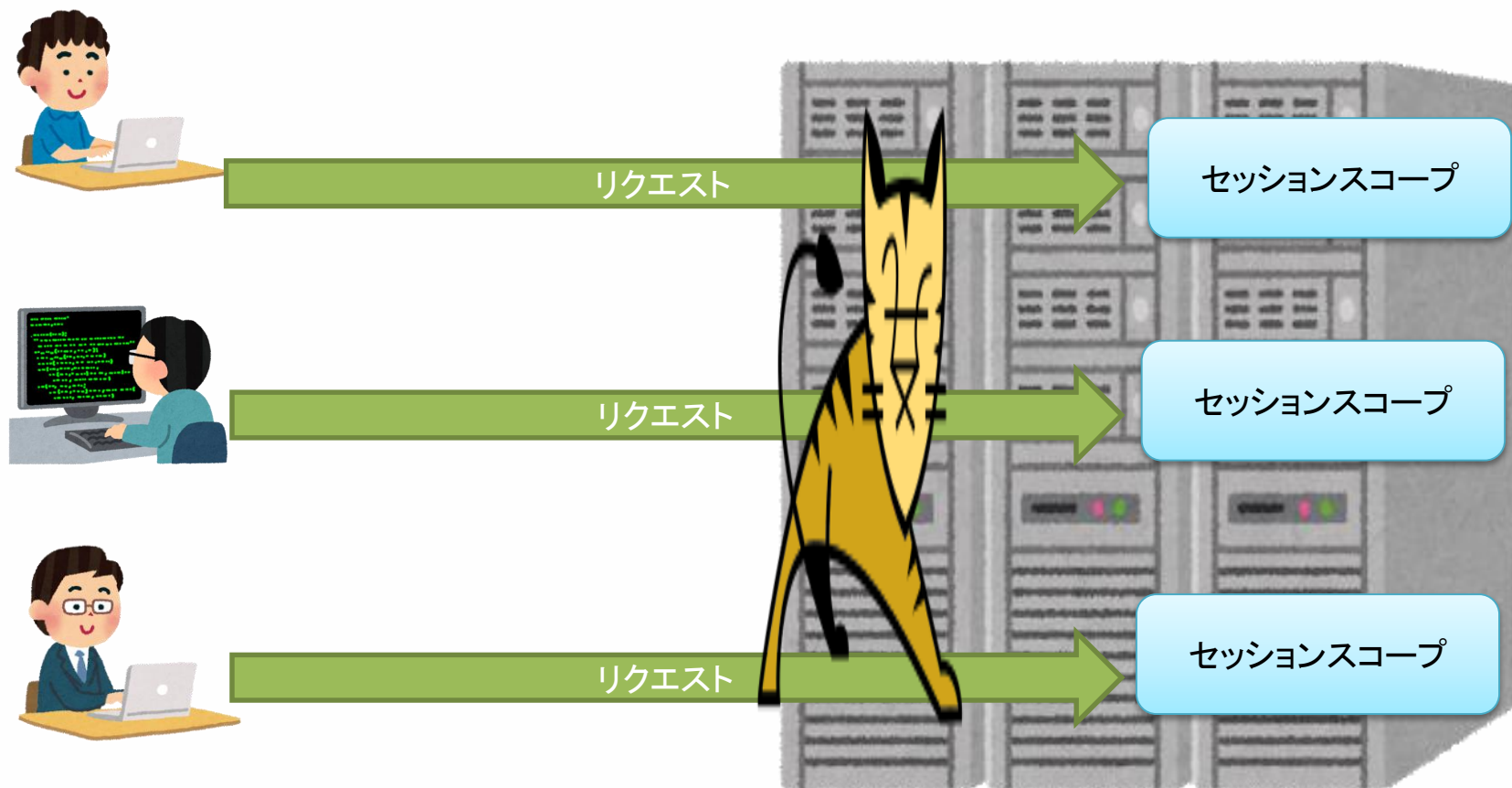
セッションにも「寿命」がありますが、その寿命は
ある人がサーバー接続して、ある人が切断するまで
です。

つまり、

セッションは人毎に作られます！

もっとセッションを知る

イメージ



ポイント

もっとセッションを知る

情報を置く＝メモリを消費する

何でもかんでもセッションに置くと、一人当たりのセッションのサイズが大きくなります。

**例えば、セッションに500バイトのデータをおいたとします
そのサイトに**

1000人同時にアクセスしたら？→500Mバイト

10000人同時にアクセスしたら？→5Gバイト

サーバーのメモリがパンクしちゃいますね・・・



ポイント

もっとセッションを知る

なお、セッションの寿命について、セッションはいつ破棄されるのでしょうか？

大まかに3つあります。

1. **ブラウザを閉じた時**
2. **プログラムで、`session.invalidate`を実行した時**
3. **セッションタイムアウトが発生した時**

A decorative blue header at the top left of the slide, featuring various white geometric shapes like circles, squares, and diamonds, some of which are nested or overlapping.

目次

セッションの復習
もっとセッションを知る
演習

やってみよう!

演習

セッションについて演習をやいましょう

<https://github.com/nishino-naoyuki/2019Web>
・サーバレット課題5.pdf



まとめ

今日の目標！

リクエストスコープ セッションスコープを理解する

- 様々な画面で**共通で使いたい値**は
リクエストスコープでは実現不可能
→セッションスコープでなら実現可能

• セッションを使用するのは3STEP

1. セッションの取得 `HttpSession session = request.getSession(true);`
2. セッションへのセット `session.setAttribute("loginName", "nishino");`
3. セッションからの取得 `(String)session.getAttribute("loginName");`