



CS544 EA

# Applications

## Spring MVC: Session

# HttpSession

- If you want you can have direct access to the HttpSession, by **requesting it as a parameter**

Creates session if it didn't exist yet, or passes existing

```
@GetMapping(value="/cars/session")
public @ResponseBody String session(HttpSession session) {
    Enumeration<String> attributes = session.getAttributeNames();
    StringBuilder output = new StringBuilder();
    while (attributes.hasMoreElements()) {
        output.append(attributes.nextElement());
        output.append(" ");
    }
    return output.toString();
}
```

# @SessionAttributes

- @Controller can specify @SessionAttributes
  - Intended for the duration of the controller
- Lists the names of **model attributes** that should be **stored in the session** (instead of request)
  - Once in the session will be inside model on each subsequent request

```
@Controller
@SessionAttributes(value={"cars", "currentId"})
public class CarController {
    ...
}
```

# Removed on Completion

```
@Controller
@SessionAttributes(value={"cars", "currentId"})
public class CarController {

    ...

    @GetMapping(value="/cars/clear")
    public String clear(SessionStatus status) {
        // clears SessionAttributes specified on classlevel
        status.setComplete();
        return "redirect:/cars";
    }

    ...
}
```

Additional Parameter that allows  
you to clear (complete) the session

# Storing / Retrieving

```
@Controller
@SessionAttributes(value={"cars", "currentId"})
public class CarController {

    @PostMapping(value="/cars")
    public String getCars(Model m) {
        m.addAttribute("cars", carDao.getCars());
        m.addAttribute("currentId", 1);
        return "redirect:/cars";
    }

    @GetMapping(value="/cars")
    public String viewCars(Model m) {
        List<Car> cars = (List<Car>)m.get("cars"); // just to demonstrate
        int num = m.get("currentId");
        return "cars";
    }
}
```

Method never explicitly uses HttpSession

Once added to the model it will be available to subsequent requests

Works if it is called after the method shown above