

ITMD 462/562

Web Site Application Development

Lecture 6

Fall 2015 – September 30, 2015

Tonight's Agenda

- Sessions & Cookies
- CRUD
- CRUD Application Demo

Sessions & Cookies

Session Basics

- Sessions are a way to store simple data for individual users based on a unique session ID.
- This can persist data between application pages.
- Session IDs are usually sent to the browser and server via cookies
- The actual session data is stored on the server
- `$_SESSION` superglobal is where the data is stored.
- `session_start()` function manually starts the session
- <http://php.net/manual/en/session.examples.basic.php>

Cookies

- Cookies are a way of storing simple text data in the users browser.
- Returned to the server that set it when a new request is sent to the server.
- Only the domain that set the cookie can read it.
- If you want to set a cookie it must be called before any output is sent to the browser.
- Cookie data is in the `$_COOKIE` superglobal
- `set_cookie()` function sets a cookie.
- <http://php.net/manual/en/features.cookies.php>
- <http://php.net/manual/en/function.setcookie.php>

CRUD

CRUD

- Create, Read, Update, Delete
- 4 Basic operations of persistent storage
- Can be implemented in numerous persistence layers
 - SQL Databases
 - Object Databases
 - XML DB, NoSQL, and other database formats
 - Flat Files
 - Custom Formats and more

CRUD

- Typically map to database or application layer operations
- Most often SQL or database language
- Also maps to specific HTTP verbs for REST model
- Fundamental operations to interact with an application and data

Operation	SQL	HTTP	DDS
Create	INSERT	PUT / POST	write
Read (Retrieve)	SELECT	GET	read / take
Update (Modify)	UPDATE	PUT / PATCH	write
Delete (Destroy)	DELETE	DELETE	dispose

Data Access

Data Access Layer

- You could write direct data access code in your application everywhere it is needed.
- This leads to a lot of locations to test and change if you data access model changes.
- One Data Access Pattern that has emerged is the Repository Pattern

Repository Pattern

- Adds a separation layer between the data mapping and the domain layers of an application and acts like an in-memory collection of domain objects.
- Objects are added and modified through the repository
- This encapsulates the specific data access methods behind the scenes.
- Helps to enforce a clean separation between data mapping and domain logic layers
- If done well, the data storage layer can be changed with minimal changes to application or domain layer logic.
- <http://martinfowler.com/eaCatalog/repository.html>
- <http://shawnmc.cool/the-repository-pattern>

Repository Pattern

- Commonly a more generic interface is defined in a repository object.
- That interface defines the methods that will need to be implemented by any class that implements that interface.
- Specific Classes will be created that implement that interface which implement the methods based on the underlying data storage method
- <http://shawnmc.cool/the-repository-pattern>

Assignments

Assignments

- Assignment 2 will be posted tonight.