CHAPTER 6

Data Objects

"I'm an idealist. I don't know where I'm going, but I'm on my way." —Carl Sandburg, Incidentals (1904)

Chapter Objectives/Student Learning Outcomes

After completing this chapter, the student will be able to:

- Create a data class that inserts, updates, and deletes XML or JSON data
- Explain how to create a data class that updates MySQL data using a SQL script
- Create a PHP program that creates a change backup log
- Create a PHP program that can recover data from a previous backup
- Apply changes to create up-to-date valid information
- Use dependency injection to attach a data class to another class in the BR tier
- Create a three-tier PHP application

The Data Class

The interface and business rules tiers should not store application information. These tiers should not even be aware of how the information is stored (text file, XML, or database) or the location of the stored information. Any information that is stored must be passed from the business rules tier to the data tier. The data tier is also responsible for reacting to requests for information from the business rules tier.

This allows the interface tier and business rules tier to be unaware of any changes in types of storage methods (text file, XML, or database) and the locations of stored items. The signature (parameters accepted) and items returned from the data tier should remain unchanged over the life of the application. As long as these do not change, there should be no changes needed in the other tiers when changes occur in the data tier.

Security and performance—When using databases it may seem logical to build a SQL string in the business rules tier and pass the string to the data tier. This would cause a major security hole in the application. Hackers could pass any SQL string (including a `delete` string). It may also seem logical to pass SQL update commands (`DELETE`, `UPDATE`, and `INSERT`) into the data tier. Again this provides a major hole. Passing data for a `WHERE` SQL command is also a bad idea as it might allow hackers to delete or change any combination of data in the database.

Electronic supplementary material The online version of this chapter (doi:10.1007/978-1-4842-1730-6_6) contains supplementary material, which is available to authorized users.

S. Prettyman, *Learn PHP 7*, DOI 10.1007/978-1-4842-1730-6_6