

Contents

GUIDED EXERCISE 1.1: SETTING UP YOUR APPLICATION DEVELOPMENT ENVIRONMENT	1
<i>Part 1—Starting the Database Server.....</i>	<i>2</i>
<i>Part 2—Setting up a workspace in Developer Studio</i>	<i>4</i>
<i>Part 3—Setting up the Server project</i>	<i>7</i>
<i>Guided Exercise 1.1: Setting up your application development environment, Wrap-up.....</i>	<i>13</i>
TRY IT 2.1: DEFINING CLASSES.....	15
<i>Part 1—Creating the Emp class.....</i>	<i>16</i>
<i>Part 2—Defining data members for the Emp class.....</i>	<i>17</i>
<i>Part 3—Defining methods for the Emp class.....</i>	<i>18</i>
<i>Part 4—Adding the include file for the ttEmployee temp-table.....</i>	<i>19</i>
<i>Part 5—Creating the Dept class.....</i>	<i>20</i>
<i>Part 6—Defining data members for the Dept class.....</i>	<i>21</i>
<i>Part 7—Defining a constructor and methods for the Dept class</i>	<i>22</i>
<i>Solution, Part 1—Creating the Emp class.....</i>	<i>23</i>
<i>Solution, Part 2—Defining data members for the Emp class.....</i>	<i>26</i>
<i>Solution, Part 3—Defining methods for the Emp class.....</i>	<i>30</i>
<i>Solution, Part 4—Adding the include file for the ttEmployee temp-table</i>	<i>35</i>
<i>Solution, Part 5—Creating the Dept class.....</i>	<i>37</i>
<i>Solution, Part 6—Defining data members for the Dept class</i>	<i>39</i>
<i>Solution, Part 7—Defining a constructor and methods for the Dept class</i>	<i>41</i>
<i>Try It 2.1: Defining classes, Wrap-up.....</i>	<i>45</i>
TRY IT 2.2: WORKING WITH CLASSES.....	47
<i>Part 1—Implementing the methods for the Emp class</i>	<i>48</i>
<i>Part 2—Implementing the methods for the Dept class.....</i>	<i>49</i>
<i>Solution, Part 1—Implementing the methods of the Emp class.....</i>	<i>50</i>
<i>Solution, Part 2—Implementing the methods for the Dept class.....</i>	<i>52</i>
<i>Try It 2.2: Working with classes, Wrap-up</i>	<i>54</i>
TRY IT 2.3: TESTING CLASSES.....	55
<i>Part 1—Setting up a Test Project.....</i>	<i>56</i>
<i>Part 2—Writing the test procedure for the Emp class</i>	<i>57</i>
<i>Part 3—Testing the Emp class</i>	<i>58</i>
<i>Part 4—Writing the test procedure for the Dept class.....</i>	<i>59</i>
<i>Part 5—Testing the Dept class.....</i>	<i>61</i>
<i>Solution, Part 1—Setting up a Test project.....</i>	<i>62</i>
<i>Solution, Part 2—Writing the test procedure for the Emp class</i>	<i>68</i>
<i>Solution, Part 3—Testing the Emp class.....</i>	<i>72</i>
<i>Solution, Part 4—Writing the test procedure for the Dept class.....</i>	<i>74</i>
<i>Solution, Part 5—Testing the Dept class</i>	<i>80</i>
<i>Try It 2.3: Testing classes, Wrap-up</i>	<i>81</i>
TRY IT 3.1: USING INHERITANCE	83
<i>Part 1—Modify the Emp class to support its derived classes.....</i>	<i>84</i>
<i>Part 2—Creating the Manager class</i>	<i>85</i>
<i>Part 3—Defining a constructor and a data member for the Manager class</i>	<i>86</i>
<i>Part 4—Defining methods for the Manager class.....</i>	<i>87</i>
<i>Part 5—Implementing the methods for the Manager class.....</i>	<i>88</i>
<i>Part 6—Importing the TeamMember and Dept classes</i>	<i>89</i>
<i>Part 7—Testing the inheritance hierarchy.....</i>	<i>90</i>
<i>Solution, Part 1—Modify the Emp class to support its derived classes</i>	<i>85</i>
<i>Solution, Part 2—Creating the Manager class</i>	<i>91</i>
<i>Solution, Part 3—Defining a constructor and a data member for the Manager class.....</i>	<i>92</i>

<i>Solution, Part 4—Defining methods for the Manager class</i>	95
<i>Solution, Part 5—Implementing the methods of the Manager class</i>	99
<i>Solution, Part 6—Importing the TeamMember and Dept classes</i>	101
<i>Solution, Part 7—Testing the inheritance hierarchy</i>	103
<i>Try It 3.1: Using inheritance, Wrap-up</i>	108
TRY IT 3.2: USING AN INTERFACE CLASS	109
<i>Part 1—Creating the IBusiness Unit interface class</i>	110
<i>Part 2—Defining data members for the IBusiness Unit interface class</i>	111
<i>Part 3—Defining methods for the IBusiness Unit interface class</i>	112
<i>Part 4—Creating the Company class</i>	113
<i>Part 5—Implementing a constructor, a destructor, and methods for the Company class</i>	114
<i>Part 6—Importing the Franchise class</i>	115
<i>Part 7—Testing the classes</i>	116
<i>Solution, Part 1—Creating the IBusiness Unit interface class</i>	117
<i>Solution, Part 2—Defining data members for the IBusinessUnit interface class</i>	119
<i>Solution, Part 3—Defining methods for the IBusinessUnit interface class</i>	121
<i>Solution, Part 4—Creating the Company class</i>	123
<i>Solution, Part 5—Implementing a constructor, a destructor, and methods for the Company class</i>	126
<i>Solution, Part 6—Importing the Franchise class</i>	128
<i>Solution, Part 7—Testing the classes</i>	129
<i>Try It 3.2: Using an interface class, Wrap-up</i>	132
TRY IT 3.3: USING A SINGLETON AND CREATING CLASSES DYNAMICALLY	133
<i>Part 1—Importing the Corporation class and the ttBusinessUnit include file</i>	134
<i>Part 2—Defining static data members for the Corporation class</i>	135
<i>Part 3—Defining a static constructor for the Corporation class</i>	136
<i>Part 4—Adding code to the InitializeBusinessUnit() method to create instances dynamically</i>	137
<i>Part 5—Testing the Corporation class</i>	138
<i>Solution, Part 1—Importing the Corporation class and the ttBusinessUnit include file</i>	139
<i>Solution, Part 2—Defining static data members for the Corporation class</i>	141
<i>Solution, Part 3—Defining a static constructor for the Corporation class</i>	143
<i>Solution, Part 4—Adding code to the InitializeBusinessUnit() method to create instances dynamically</i>	144
<i>Solution, Part 5—Testing the Corporation class</i>	145
<i>Try It 3.3: Using a singleton and creating classes dynamically, Wrap-up</i>	146
TRY IT 3.4: USING EVENTS	147
<i>Part 1—Defining and publishing an event in the Manager class</i>	148
<i>Part 2—Modify the Dept class to subscribe to the event</i>	149
<i>Part 3—Testing the Dept class event</i>	150
<i>Solution, Part 1—Defining and publishing an event in the Manager class</i>	151
<i>Solution, Part 2—Modify the Dept class to subscribe to the event</i>	154
<i>Solution, Part 3—Testing the Dept class event</i>	157
<i>Try It 3.4: Using events, Wrap-up</i>	158