File - C:\Users\ctqdt\IdeaProjects\CourseScheduler\src\com\company\data\person\PersonName.java

```java
/**
 * @author Caroline Ta
 * @since 05.19.2020
 */
package com.company.data.person;

import org.jetbrains.annotations.NotNull;

/**
 * The type Person name.
 */
public class PersonName implements Comparable<PersonName> {

    /**
     * The Person first name.
     */
    private String firstName;
    /**
     * The Person middle name.
     */
    private String midName;
    /**
     * The Person last name.
     */
    private String lastName;

    // ------------------------------------------------------------------------
    // CONSTRUCTORS
    // ------------------------------------------------------------------------

    /**
     * Instantiates a new Person name.
     */
    public PersonName() {
    }

    /**
     * Instantiates a new Person name.
     *
     * @param firstName the first name
     * @param midName   the mid name
     * @param lastName  the last name
     * @throws Exception the exception
     */
    public PersonName(String firstName, String midName, String lastName) throws Exception {
        setFirstName(firstName);
        setMidName(midName);
        setLastName(lastName);
    }

    // ------------------------------------------------------------------------
    // ACCESSORS - GETTER METHODS
    // ------------------------------------------------------------------------

    /**
     * Gets first name.
     *
     * @return the first name
     */
    public String getFirstName() {
        return firstName;
    }

    /**
     * Gets mid name.
     *
     * @return the mid name
     */
    public String getMidName() {
        return midName;
```

```
71      }
72
73      /**
74       * Gets last name.
75       *
76       * @return the last name
77       */
78      public String getLastName() {
79          return lastName;
80      }
81
82      // ------------------------------------------------------------------------
83      // MUTATORS - SETTER METHODS
84      // ------------------------------------------------------------------------
85
86      /**
87       * Sets first name.
88       *
89       * @param firstName the first name
90       * @throws Exception the exception
91       */
92      public void setFirstName(String firstName) throws Exception {
93          checkStringValue(firstName, "Firstname");
94          this.firstName = firstName;
95      }
96
97      /**
98       * Sets mid name.
99       *
100      * @param midName the mid name
101      * @throws Exception the exception
102      */
103     public void setMidName(String midName) throws Exception {
104         checkStringValue(firstName, "Midname");
105         this.midName = midName;
106     }
107
108     /**
109      * Sets last name.
110      *
111      * @param lastName the last name
112      * @throws Exception the exception
113      */
114     public void setLastName(String lastName) throws Exception {
115         checkStringValue(firstName, "Lastname");
116         this.lastName = lastName;
117     }
118
119     // ------------------------------------------------------------------------
120     // FUNCTIONALITY METHODS
121     // ------------------------------------------------------------------------
122
123     /**
124      * Check string value.
125      *
126      * @param valueString   the value string
127      * @param attributeName the attribute name
128      * @throws Exception the exception
129      */
130     private void checkStringValue(String valueString, String attributeName) throws Exception {
131         if (valueString == null || valueString.trim().equals("")) {
132             throw new Exception(attributeName + " must be non-null and non-empty");
133         }
134     }
135
136     // ------------------------------------------------------------------------
137     // OVERRIDDEN METHODS
138     // ------------------------------------------------------------------------
139
140     /**
```

```
141      * Override toString method.
142      *
143      * @return a string description for the PersonName class.
144      */
145     @Override
146     public String toString() {
147         return firstName + " " + midName + " " + lastName;
148     }
149
150     /**
151      * Override compareTo method.
152      *
153      * @param otherPerson other person name
154      * @return comparision result (i.e. -1, 0) between PersonNames.
155      */
156     @Override
157     public int compareTo(@NotNull PersonName otherPerson) {
158         if (this.firstName.equalsIgnoreCase(otherPerson.firstName)
159                 && this.midName.equalsIgnoreCase(otherPerson.midName)
160                 && this.lastName.equalsIgnoreCase(otherPerson.lastName)) {
161             return 0;
162         } else {
163             return -1;
164         }
165     }
166
167     /**
168      * Override equals method.
169      *
170      * @param obj object
171      * @return the result of thisPersonName equals otherPersonName as a boolean
172      */
173     @Override
174     public boolean equals(Object obj) {
175         if (obj == null || !(obj instanceof PersonName)) {
176             return false;
177         }
178
179         return compareTo((PersonName) obj) == 0;
180     }
181 }
182
```