

ETAPE 7 : création d'une nouvelle étape

Ajout d'une nouvelle étape :

On procède presque de la même façon que lorsque qu'on ajoute une excursion mais cette fois ci, on le fait pour la table Etape :

```
private void ajouterEtape(String code, String nom, int duree, String description) {
    // SQL pour insérer une nouvelle étape
    String sql = "INSERT INTO Etape (CodeExcursion, NumEtape, DureePrevue, Description) "
        + "VALUES (?, ?, ?, ?)";

    try (Connection conn = ConnexionBDD.getConnexion();
        PreparedStatement stmt = conn.prepareStatement(sql)) {

        // Récupérer le dernier NumEtape pour l'excursion donnée et l'incrémenter
        String selectLastNumEtape = "SELECT MAX(NumEtape) FROM Etape WHERE CodeExcursion = ?";
        try (PreparedStatement stmtSelect = conn.prepareStatement(selectLastNumEtape)) {
            stmtSelect.setString(1, code);
            ResultSet rs = stmtSelect.executeQuery();

            int numEtape = 1; // Valeur par défaut si aucune étape n'existe pour l'excursion
            if (rs.next()) {
                numEtape = rs.getInt(1) + 1; // Incrémenter le dernier numéro d'étape
            }

            // Préparer l'insertion de l'étape
            stmt.setString(1, code); // Code de l'excursion
            stmt.setInt(2, numEtape); // Numéro de l'étape
            stmt.setInt(3, duree); // Durée de l'étape
            stmt.setString(4, description); // Description de l'étape

            stmt.executeUpdate();
        }
    } catch (SQLException e) {
        JOptionPane.showMessageDialog(parentComponent, this, "Erreur lors de l'ajout : " + e.getMessage(), title: "Erreur", messageType: JOptionPane.ERROR_MESSAGE);
    }
}
```

Le bouton OK fonctionne de la même manière et exécute la requête SQL qui crée la ligne dans la table Etape avec les données qui ont été rentrées

```
private void jButtonOkActionPerformed(java.awt.event.ActionEvent evt) {
    // Récupération des valeurs des champs
    String code = jTextFieldCode.getText().trim();
    String nom = jTextFieldNom.getText().trim();
    String dureeStr = jTextFieldDuree.getText().trim();
    String description = jTextFieldDescription.getText().trim();

    // Vérification que tous les champs sont remplis
    if (code.isEmpty() || nom.isEmpty() || dureeStr.isEmpty() || description.isEmpty()) {
        JOptionPane.showMessageDialog(parentComponent, this, "Veuillez remplir tous les champs", title: "Erreur", messageType: JOptionPane.ERROR_MESSAGE);
        return;
    }

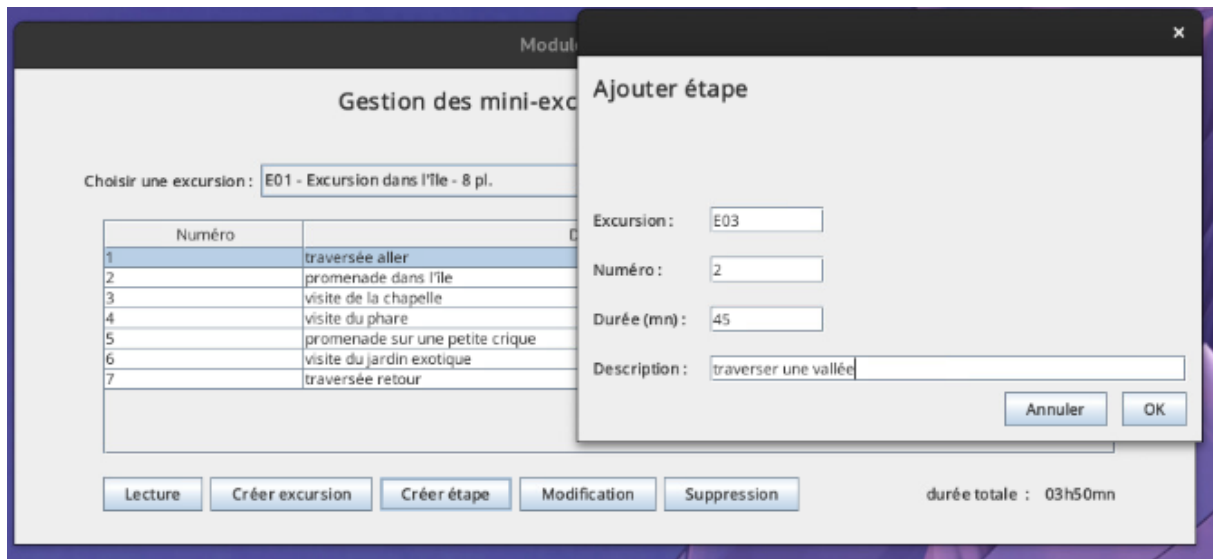
    try {
        int duree = Integer.parseInt(dureeStr); // Conversion de la durée

        // Appel à la méthode pour ajouter l'étape
        ajouterEtape(code, nom, duree, description);

        // Message de confirmation
        JOptionPane.showMessageDialog(parentComponent, this, "Étape ajoutée avec succès", title: "Succès", messageType: JOptionPane.INFORMATION_MESSAGE);
    } catch (NumberFormatException e) {
        JOptionPane.showMessageDialog(parentComponent, this, "La durée doit être un nombre valide", title: "Erreur", messageType: JOptionPane.ERROR_MESSAGE);
    }

    this.dispose(); // Fermeture de la fenêtre après ajout
}
```

Fonctionnement de l'ajout d'étape :



Lorsque les données sont entrées correctement et que l'on clique sur OK, un message précise si l'insertion a bien fonctionné. En vérifiant dans la BDD, on observe que la nouvelle ligne apparaît tout comme dans la fenêtre JFRAME dans la bonne excursion.

