

Explications et code correspondant à l'enregistrement des informations relatives à un bateau qui vient de s'amarrer

Affichage de toutes les informations sur le bateau

Emplacement :

Nom du bateau :

Pavillon : 

Port d'attache :

Tonnage :

Est submersible :

Energie :

Infos. supplémentaires :

Mon objet Equipage est créé lorsque je clique sur le bouton amarré en ayant sélectionné un bateau dans la liste des bateaux en entrées.

```
equipage = new Equipage ();
```

```
private void BoutonEquipageActionPerformed(java.awt.event.ActionEvent evt) {  
    java.awt.EventQueue.invokeLater(() -> {  
        new PublishEquipage(equipage).setVisible(true);  
    });  
}
```

Lorsque je clique sur le bouton Equipage, je lance une dialog en passant en paramètre mon objet équipage

Fenêtre d'enregistrement d'un équipage

Bateau :

Nom :

Prénom :

Date de naissance :

Registre national :

Nationalité :

☐ Capitaine
 ☐ Second
 ☐ Bosco
 ☐ Matelot
 ☐ Passager

Numéro de registre n...	Nom	Prénom	Date de naissance	Nationalité	Fonction

Voici la Dialog pour insérer des marins dans mon objet équipage

Quand j'ajoute un membre, je récupère chaque TextBox :

```
private void BoutonAjouterActionPerformed(java.awt.event.ActionEvent evt) {

    String RegistreNationalMarin = LabelRegistre.getText();
    String Nom = LabelNom.getText();
    String Prenom = LabelPrenom.getText();
    String DateNaissance = BoxJour.getSelectedItem() + "/" +
BoxMois.getSelectedItem().toString() + "/" + BoxAnnee.getSelectedItem().toString();
    String Nationalite = LabelNationalite.getText().toUpperCase();
    String Fonction = null;
    if (BoxCapitaine.isSelected()){
        Fonction = "Capitaine";
    }
    else if (BoxSecond.isSelected()) {
        Fonction = "Second";
    }
    else if (BoxBosco.isSelected()) {
        Fonction = "Bosco";
    }
    else if (BoxMatelot.isSelected()) {
        Fonction = "Matelot";
    }
    else if (BoxPassager.isSelected()) {
        Fonction = "Passager";
    }
    else {
        Fonction = "Clandestin";
    }
}
```

```

if (TableEquipage.getRowCount() > 0) // Si un membre existe déjà
{
    for (int j = 0; j < TableEquipage.getRowCount(); j++)
    {
        if (TableEquipage.getValueAt(j, 0).equals(RegistreNationalMarin)) // Je vérifie
seulement pour les registres de l'équipage actuel (à modifier)
        {
            String msg;
            msg = "Un membre de l'équipe a déjà ce numéro de registre !";
            JOptionPane.showMessageDialog(this, msg, "Erreur de format",
JOptionPane.WARNING_MESSAGE);
            return;
        }
        if (TableEquipage.getValueAt(j, 5).equals("Second") && Fonction.equals("Second"))
        {
            String msg;
            msg = "Vous avez déjà ajouté un second !";
            JOptionPane.showMessageDialog(this, msg, "Attention !",
JOptionPane.INFORMATION_MESSAGE);
            return;
        }
        if (TableEquipage.getValueAt(j, 5).equals("Capitaine") &&
Fonction.equals("Capitaine")) // Un capitaine existe déjà
        {
            String msg;
            msg = "Vous avez déjà ajouté un capitaine !";
            JOptionPane.showMessageDialog(this, msg, "Attention !",
JOptionPane.INFORMATION_MESSAGE);
            return;
        }
    }
    try
    {
        Date dateNaissance=new SimpleDateFormat("dd/MM/yyyy").parse(DateNaissance);
        try
        {
            Marin marin = new Marin (RegistreNationalMarin, Nom, Prenom, Nationalite,
dateNaissance, Fonction);
            Vector ligne = new Vector();
            ligne.add(marin.getIdentifiant());
            ligne.add(marin.getNom());
            ligne.add(marin.getPrenom());
            ligne.add(DateNaissance);
            ligne.add(Nationalite);
            ligne.add(marin.getFonction());
            Capitainerie.fichierLog.ecritLigne ("Création du marin : " +
RegistreNationalMarin);
            DTMEquipage.addRow(ligne);
            this.TableEquipage.setModel(DTMEquipage);
        } catch (SailorWithoutIdentificationException ex) {
            Capitainerie.fichierLog.ecritLigne ("SailorWithoutIdentificationException : "
        }
    } catch (ParseException e){
        String msg;
        msg = "Erreur : la date est invalide !";
        JOptionPane.showMessageDialog(this, msg, "Erreur de format",
JOptionPane.WARNING_MESSAGE);
    }
}
}

```

Je vérifie d'abord si la liste d'équipage est vide, si oui, l'utilisateur doit obligatoirement ajouter un capitaine. S'il a sélectionné la box Capitaine, j'ajoute le capitaine. Si non, j'ouvre un MessageDialog avec le message. Si il a sélectionné un second, je vérifie si il n'en existe pas déjà un etc.

Si la liste n'est pas null, c'est que nous avons déjà notre capitaine, lors d'un autre ajout, je vérifie d'abord le numéro de registre est unique (PK) au sein de la liste (à modifier). Ensuite, je vérifie si la Box qu'il a sélectionné n'est pas celle du capitaine. (Capitaine unique, Second unique).

```
else // Création du capitaine
{
    if (Fonction.equalsIgnoreCase("Capitaine")) // Je vérifie si c'est bien un
    capitaine que l'user veut ajouter
    {
        try
        {
            Date dateNaissance = new
SimpleDateFormat("dd/MM/yyyy").parse(DateNaissance);
            try {
                Marin marin = new Marin (RegistreNationalMarin, Nom, Prenom,
Nationalite, dateNaissance, Fonction);
                Vector ligne = new Vector();
                ligne.add(marin.getIdentifiant());
                ligne.add(marin.getNom());
                ligne.add(marin.getPrenom());
                ligne.add(DateNaissance);
                ligne.add(Nationalite);
                ligne.add(marin.getFonction());
                Capitainerie.fichierLog.ecritLigne ("Création du capitaine : " +
RegistreNationalMarin);
                DTMEquipage.addRow(ligne);
                this.TableEquipage.setModel(DTMEquipage);
            } catch (SailorWithoutIdentificationException ex) {
                Capitainerie.fichierLog.ecritLigne
("SailorWithoutIdentificationException : " + ex.getMessage());
            }
        } catch (ParseException e){
            String msg;
            msg = "Erreur : la date est invalide !";
            JOptionPane.showMessageDialog(this, msg, "Erreur de format",
JOptionPane.WARNING_MESSAGE);
        }
    }
    else
    {
        String msg;
        msg = "Vous devez ajouter un capitaine comme premier membre d'équipage
!";
        JOptionPane.showMessageDialog(this, msg, "Attention !",
JOptionPane.WARNING_MESSAGE);
    }
}
```

Création d'un capitaine

```

DefaultComboBoxModel DCBMEquipage = new DefaultComboBoxModel();
for(int i = 0; i < TableEquipage.getRowCount(); i++) {
    try {
        String elem = TableEquipage.getValueAt(i, 5) + " : " +
        TableEquipage.getValueAt(i, 2) + " " + TableEquipage.getValueAt(i, 1); // Rajoute dans la
        liste "Capitaine : Prénom Nom"
        DCBMEquipage.addElement(elem);

        // On remplit l'équipage.
        String RegistreNational = (String) TableEquipage.getValueAt(i, 0);
        String Nom = (String) TableEquipage.getValueAt(i, 1);
        String Prenom = (String) TableEquipage.getValueAt(i, 2);
        String DateNaissance = (String) TableEquipage.getValueAt(i, 3);
        String Nationalite = (String) TableEquipage.getValueAt(i, 4);
        try
        {
            Date dateNaissance=new SimpleDateFormat("dd/MM/yyyy").parse(DateNaissance);

            String Fonction = (String) TableEquipage.getValueAt(i, 5);
            switch (Fonction){
                case "Capitaine":
                    Marin Capitaine = new Marin (RegistreNational, Nom, Prenom,
                    Nationalite, dateNaissance, Fonction);
                    this.equipage.ajouterMarin(Capitaine);
                    Capitainerie.fichierLog.ecritLigne("Création d'un capitaine : " +
                    RegistreNational);
                    break;
                case "Second":
                    Marin Second = new Marin (RegistreNational, Nom, Prenom, Nationalite,
                    dateNaissance, Fonction);
                    this.equipage.ajouterMarin(Second);
                    Capitainerie.fichierLog.ecritLigne("Création d'un second : " +
                    RegistreNational);
                    break;
                default:
                    Marin mar = new Marin (RegistreNational, Nom, Prenom, Nationalite,
                    dateNaissance, Fonction);
                    this.equipage.ajouterMarin(mar);
                    Capitainerie.fichierLog.ecritLigne("Création d'un marin : " +
                    RegistreNational);
                    break;
            }
        }catch (ParseException e){
            String msg;
            msg = "Erreur : la date est invalide !";
            JOptionPane.showMessageDialog(this, msg, "A propos",
            JOptionPane.INFORMATION_MESSAGE);
        }
    } catch (SailorWithoutIdentificationException ex) {
        Capitainerie.fichierLog.ecritLigne ("SailorWithoutIdentificationException : " +
        ex.getMessage());
    }
}
InfosBateau.ListEquipage.setModel((ComboBoxModel) DCBMEquipage);
setVisible(false);
this.TableEquipage.clearSelection();

```

Lorsque j'ai fini d'ajouter, modifier, (supprimer) mon équipage. Je le valide. C'est alors que pour chaque ligne de ma JTable, je récupère chaque champ et je crée un objet Marin que je vais ajouter à mon équipage lié au bateau.

```
bateauPlaisance = new BateauPlaisance (IdentifiantBateau, SubmersibleBateau,  
EnergieBateau, NomBateau, PortAttacheBateau, TonnageBateau, LongueurBateau,  
PavillonBateau, equipage, Supplement);|
```

Dans ma fenêtre infosBateaux, je crée un objet BateauPeche/BateauPlaisance et j'insère en tant que paramètre, l'équipage actuel.