

Section : N° d'inscription : Série :

Nom et Prénom :

Date et lieu de naissance :

Signature des
surveillants



Epreuve : STI - Section : Sciences de l'informatique – Session de contrôle 2023

Le sujet comporte 15 pages numérotées de 1 sur 15 à 15 sur 15.

Les pages de 1 à 11 sont à remettre.

20

Partie A : Gestion de la base de données

(12 points)

En se référant à la représentation textuelle de la base de données "Gestion_Incendies" et au tableau de description des champs figurants à la page 12, répondre aux questions des exercices qui suivent.

Exercice 1 : Interprétation de la base de données

(3,25 points)

- 1) Compléter la colonne "Valide" du tableau ci-dessous par le mot "Oui" si l'affirmation proposée est valide ou par le mot "Non" dans le cas contraire.

Affirmation	Valide
Chaque gouvernorat comporte une ou plusieurs villes, mais chaque ville est rattachée à un seul gouvernorat.
Une intervention peut concerner plusieurs incendies simultanément.
Pour un type d'intervention, on peut mener plusieurs intervention en même temps.
Il est possible que le statut d'un incendie soit éteint sans nécessiter aucune intervention.
La date et l'heure d'extinction est obligatoire lors de l'insertion des données dans la table "Incendie".
Un incendie peut nécessiter plusieurs interventions en même temps.
Une forêt peut être rattachée à plusieurs villes.
Le même incendie peut se produire dans plusieurs forêts.

- 2) Pour chacune des affirmations suivantes, mettre une croix (X) dans la case correspondante à la réponse correcte.

Important : Pour chaque affirmation, toute réponse comportant *plus d'une croix* est considérée *erronée*.

- a) Pour initialiser le champ "statutInc" à la lettre "A", lors de la création de la table "Incendie", on utilise :

☐ DEFAULT

☐ DISTINCT

☐ NULL

- b) Pour créer une relation entre la table "Gouvernorat" et la table "Ville", on utilise :

☐ PRIMARY KEY

☐ NOT NULL

☐ REFERENCES

- c) Pour obliger l'utilisateur à saisir un contenu dans le champ "nomForêt", on utilise :

☐ DEFAULT

☐ UNIQUE

☐ NOT NULL

Ne rien écrire ici

d) Pour garantir que le contenu du champ "nomGouv" ne se répète pas, on utilise :

☐ CHECK

☐ UNIQUE

☐ REFERENCES

e) Pour supprimer un enregistrement de la table "Ville" ainsi que tous les enregistrements de la table "Foret" qui lui sont associés, on utilise :

☐ ON DELETE CASCADE

☐ ON UPDATE CASCADE

☐ ON DROP CASCADE

f) Pour ajouter une contrainte d'intégrité nommée "FKTypeInt" sur la clé étrangère "codeTypeInt" de la table "Intervention", on utilise la commande :

☐ DROP TABLE

☐ ALTER TABLE

☐ UPDATE

3) On donne ci-dessous un extrait des données à insérer dans les tables "Ville" et "Foret" :

N° ligne	Ville			Foret			
	codeVille	nomVille	codeGouv	codeForet	nomForet	superficieForet	codeVille
1	V0001	Hammam Lif	G03	F00001	Parc national de Boukornine	1 939.000	V0001
2	V0002	Bargou	G11	F00002	Forêt de Jbel Serj	1 720.000	V0002
3	V0003	Zaghouan	G06	F00003	Forêt de Jbel Zaghouan	2 010.000	V00003
4	V0004	Aïn Draham	G09	F00004	Réserve naturelle de Aïn Zana	-47.000	V0004
5	V0005	Souk Jomaa	G09	F00005	Forêt Bent Hmed	1 541.000	V0005
6	V0006	Ghardimaou	G09	F00006	Parc national d'El Feija	2 632.000	V0006
7	V0006	Nasrallah	G13	F00007	La Tourbière de Dar Fatma	15.000	V0004

Lors de l'insertion de ces données, le SGBD génère des erreurs au niveau de quelques lignes. Compléter le tableau ci-dessous en indiquant pour chaque erreur le numéro de la ligne, le nom de la table, le nom du champ ainsi que le nom de la contrainte d'intégrité non respectée (Intégrité de table (clé primaire), Intégrité référentielle, Intégrité de domaine), sachant que tous les codes des gouvernorats donnés dans la colonne "codeGouv" de la table "Ville" existent déjà dans la table "Gouvernorat".

Numéro de la ligne	Nom de la table	Nom du champ	Nom de la contrainte non respectée
.....
.....
.....

Exercice 2 : Interprétation des requêtes

(1,75 point)

On donne ci-dessous un extrait des tables "TypeIntervention" et "Intervention".

codeTypeInt	description
TI1	Aérienne
TI2	Terrestre
TI3	Maritime
TI4	Mixte

Ne rien écrire ici

codeInt	dateHeureDebInt	dateHeureFinInt	superficieInt	codeTypeInt	codeInc
INT00000001	01/07/2024 17:00	02/07/2024 12:00	80.000	TI2	INC00000001
INT00000002	02/07/2024 8:00	02/07/2024 12:00	20.000	TI1	INC00000001
INT00000003	02/07/2024 12:00	03/07/2024 18:00	100.000	TI2	INC00000001
INT00000004	05/07/2024 11:00	05/07/2024 17:00	70.000	TI4	INC00000002
INT00000005	20/08/2024 11:30	21/08/2024 12:00	200.000	TI1	INC00000003
INT00000006	21/08/2024 11:00	22/08/2024 12:00	100.000	TI2	INC00000003

1) En se basant sur l'extrait fourni, compléter le tableau ci-dessous par le résultat de l'exécution de la requête **R** suivante, tout en indiquant les noms des colonnes à afficher :

```
SELECT description, SUM(superficieInt) AS SuperficieTotale
FROM TypeIntervention T, Intervention I
WHERE T.codeTypeInt = I.codeTypeInt
GROUP BY description ;
```


2) En déduire le rôle de la requête R.

3) Réécrire la requête **R** en apportant les modifications nécessaires pour afficher uniquement les interventions ayant une superficie totale supérieure ou égale à 120.000 hectares.

Exercice 3 : Manipulation de la structure et des données de la base (5,5 points)

1) Pour chacune des affirmations suivantes, mettre une croix (X) dans la case correspondante à la réponse correcte.

Important : Pour chaque affirmation, toute réponse comportant *plus d'une croix* est considérée *erronée*.

a) On se propose de supprimer de la table "Ville" le champ "nbForet". La requête SQL permettant la suppression de ce champ est :

- ☐ ALTER TABLE Ville DELETE nbForet
☐ ALTER TABLE Ville DROP nbForet
☐ ALTER TABLE Ville DROP CONSTRAINT nbForet

b) Pour attribuer la valeur NULL au champ "dateHeureExt" de la table "Incendie" pour tous les incendies ayant le statut actif, on utilise la requête SQL :

- ☐ ALTER Table Incendie ADD dateHeureExt DATETIME NULL DEFAULT
☐ UPDATE Incendie ADD dateHeureExt DEFAULT NULL WHERE StatutInc='A'
☐ UPDATE Incendie SET dateHeureExt =NULL WHERE StatutInc='A'

c) Pour modifier le nom du champ "dateHeureExt" de la table "Incendie" par "dateHExt", on utilise la requête SQL :

- ☐ ALTER TABLE Incendie RENAME dateHeureExt To dateHExt
☐ ALTER Incendie RENAME dateHeureExt To dateHExt
☐ UPDATE TABLE Incendie SET dateHeureExt = dateHExt

d) Pour augmenter la taille du champ "nomForet" de la table "Foret" de 50 à 100 caractères, on utilise la requête SQL :

- ☐ ALTER TABLE Foret CHANGE nomForet varchar(100)
☐ ALTER TABLE Foret MODIFY nomForet varchar(100)
☐ ALTER TABLE Foret MODIFY nomForet varchar(50) To nomForet varchar(100)

2) Écrire une requête permettant d'ajouter une nouvelle intervention **non achevée**, ayant les détails suivants :

- Code de l'incendie : **INC0000010**
- Code de l'intervention : **INT0000017**
- Code du type de l'intervention : **TI4**
- Date et heure de début de l'intervention : **2025-06-01 14:00**
- Superficie de l'intervention : **8.000 hectares**

N.B. : L'incendie de code "INC0000010" et le type de l'intervention de code "TI4" existent déjà respectivement dans la table "Incendie" et la table "TypeIntervention".

Section : N° d'inscription : Série :
Nom et Prénom :
Date et lieu de naissance :

Signature des
surveillants

Epreuve : STI - Section : Sciences de l'informatique – Session de contrôle 2025

3) Écrire en SQL les requêtes permettant de répondre aux questions suivantes :

a) Quelles sont les forêts localisées dans le gouvernorat de **Jendouba** ? On affichera les champs "nomForet" et "superficieForet".

b) Quel est le nombre d'incendies enregistrés pour chaque forêt au cours de l'année **2024** ? On affichera les champs "nomForet" ainsi que le nombre d'incendies de chaque forêt.

c) Quelle est la somme des superficies brûlées pendant l'été **2023** (Juin, Juillet, Août) pour chaque ville ? On affichera les champs "nomVille" ainsi que la somme des superficies brûlées.

d) Quels sont les incendies ayant fait l'objet d'une intervention "**Aérienne**" ? On affichera, sans répétition, les champs "codeInc" et "dateHeureInc".

e) Quels sont les incendies ayant subi trois interventions ou plus ? on affichera les champs "codeInc", "nomForet" et "dateHeureInc" ainsi que le nombre d'interventions.

Ne rien écrire ici

- f) Quelles sont les forêts ayant la plus grande superficie ? on affichera les champs "nomForet" et "superficieForet".

Exercice 4 : Extension et modification de la structure de la base de données (1,5 point)

Important : À chaque représentation textuelle demandée, indiquer la **clé primaire** et la (les) **clé(s) étrangère(s)** de chaque table.

Afin d'évaluer l'impact écologique des incendies, le bureau d'études décide d'ajouter une extension à la base de données "Gestion_Incendies". Cette extension vise à gérer les espèces végétales et animales affectées par les incendies.

Cette extension consiste à ajouter les deux tables suivantes :

- **Espec** : contient des informations sur les espèces végétales et animales présentes dans les zones touchées par les incendies.
- **ImpactEspec** : documente l'impact des incendies sur les espèces enregistrées, notamment en terme de population affectée.

L'implémentation de ces deux nouvelles tables, "Espec" et "ImpactEspec", est décrite par les requêtes SQL suivantes :

```
CREATE TABLE Espec (  
  codeEspec INT PRIMARY KEY AUTO_INCREMENT,  
  nomEspec VARCHAR (50) NOT NULL,  
  typeEspec VARCHAR (8) NOT NULL CHECK (typeEspec IN ('Végétale', 'Animale'))  
);
```

```
CREATE TABLE ImpactEspec (  
  codeImpact INT PRIMARY KEY AUTO_INCREMENT,  
  codeEspec INT NOT NULL REFERENCES Espec (codeEspec),  
  codeIncendie VARCHAR(10) NOT NULL REFERENCES Incendie (codeInc),  
  populationAffectee INT  
);
```

- 1) En se basant sur les requêtes d'implémentation ci-dessus, donner la représentation textuelle des tables "Espec" et "ImpactEspec".

2) Soit les règles de gestion suivantes :

R1 : Une espèce peut être touchée par plusieurs incendies.

R2 : Un incendie peut affecter plusieurs espèces.

R3 : L'impact d'un incendie sur une espèce doit être enregistré une seule fois.

L'implémentation de la table **"ImpactEspèce"** ne respecte pas la règle de gestion **R3** car sa requête de création déjà proposée autorise l'enregistrement de l'impact sur une espèce plusieurs fois pour le même incendie. Donner une nouvelle représentation textuelle de cette table en apportant les modifications nécessaires afin de respecter les règles de gestion **R1**, **R2** et **R3**.

.....

.....

3) Pour suivre le statut des espèces touchées par les incendies (protégée, en voie de disparition, etc.), le concepteur de la base de données a décidé d'ajouter une nouvelle table nommée **"StatutEspèce"**, qui est identifiée par un champ appelé **"codeStatut"** et qui contient un autre champ appelé **"libelle"**. Cette table respecte les règles de gestion suivantes :

R4 : Une espèce peut avoir un seul statut.

R5 : Pour un statut, on peut avoir plusieurs espèces.

Donner la représentation textuelle de la nouvelle table **"StatutEspèce"** et de la table concernée par ces modifications afin de répondre aux deux règles de gestion **R4** et **R5**.

.....

.....

.....

Partie B : Programmation Web

(8 points)

Exercice 1 : Analyse et interprétation du code HTML

(3 points)

La page **"Statistiques_Interventions.html"**, dont l'aperçu est présenté dans la figure **"Fig.1"** et son code source incomplet présenté dans la figure **"Fig.2"** (page 13), permet à l'utilisateur de consulter le nombre d'interventions pour le(s) type(s) d'intervention(s) sélectionné(s) pendant une période donnée. En se basant sur ces deux figures, répondre aux questions suivantes :

1) Pour chacune des propositions ci-dessous, répondre par la lettre (V) si la proposition est correcte ou la lettre (F) dans le cas contraire.

Important : Une case vide ou contenant des lettres autres que (V) ou (F) ou des symboles est considérée comme une réponse **erronée**.

a) La balise **"<title>"** utilisée au niveau de la ligne 4 du code HTML de la figure **"Fig.2"** affiche un texte indiquant :

- ☐ le titre d'un tableau
- ☐ le titre de la page Web
- ☐ l'auteur de la page Web

b) La balise "<header>" utilisée au niveau de la ligne 9 du code HTML de la figure "Fig.2" :

- ☐ définit un conteneur pour le contenu d'introduction
☐ définit le contenu principal d'un document
☐ doit figurer dans le corps de la page web

c) Les balises (voir lignes 13, 14 et 15 du code HTML de la figure "Fig.2") qui définissent des liens hypertextes externes sont :

- ☐ Accueil
☐ Ajout des données
☐ Statistiques

d) L'attribut "required" utilisé au niveau de la ligne 22 du code HTML de la figure "Fig.2" :

- ☐ désactive la saisie dans un champ
☐ initialise le champ à une valeur donnée
☐ exige la saisie d'une valeur dans un champ

2) Pour compléter le code HTML de la figure "Fig.2" (page 13) par les balises présentes dans le tableau ci-dessous, remplir la colonne "N° Ligne" en écrivant devant chaque balise le numéro de la ligne dans laquelle elle doit être insérée.

Balise	N° Ligne
<input class="bouton" id="bt2" name="bt2" type="reset" value="Annuler">
<h2>Statistiques des Incendies</h2>
<footer id="stat">Gestion des Incendies - GeoIncendie 2025</footer>
<label>Type(s) d'intervention: </label>
<body onload="date(); temps()">
<link rel="stylesheet" href="mesStyles.css">

Exercice 2 : Analyse et interprétation du code CSS

(1 point)

1) Pour chacune des affirmations ci-dessous et en se basant sur les figures "Fig.1" et "Fig.2" (page 13) et la figure "Fig.3" (page 14), mettre une croix (X) dans la case correspondante à la réponse correcte.

Important : Pour chaque affirmation, toute réponse comportant *plus d'une croix* est considérée *erronée*.

a) Pour afficher les liens "Accueil", "Ajout des données" et "Statistiques" sur la même ligne avec une marge externe droite de 15 pixels, la règle à ajouter au niveau de la ligne 1 du code CSS (Fig.3) est :

- ☐ .menu { display: inline; margin: 0px 15px 0px 0px; }
☐ #menu { display: inline; margin: 0px 15px 0px 0px; }
☐ .menu { display: none; margin: 0px 15px 0px 0px; }

Section : N° d'inscription : Série :
 Nom et Prénom :
 Date et lieu de naissance :

Signature des
surveillants

Epreuve : **STI** - Section : **Sciences de l'Informatique** – Session de contrôle 2025

- b) Pour colorer en blanc les textes des liens "Accueil", "Ajout des données" et "Statistiques", la règle à ajouter au niveau de la ligne 2 du code CSS (*Fig.3*) est :

☐ div { color: white; }
☐ href { color: white; }
☐ a { color: white; }

- c) La propriété CSS "**background-size: 30px 30px;**", à la ligne 9 du code CSS (*Fig.3*), permet de :

☐ définir la largeur et la hauteur à 30 pixels des deux boutons
☐ définir la largeur et la hauteur à 30 pixels des images d'arrière-plan des deux boutons
☐ placer les images à 30 pixels du bord supérieur et du bord inférieur des deux boutons

- d) Pour que l'image "**valider.png**" s'affiche en tant qu'image de fond du bouton "Envoyer", il faut ajouter à la ligne 21 du code CSS (*Fig.3*) la propriété suivante :

☐ img
☐ list-style-image
☐ background-image

- 2) Quel est le rôle de la règle CSS "**.bouton:hover { text-transform: uppercase; }**" située à la ligne 12 du code CSS (*Fig.3*) ?

Exercice 3 : Analyse et interprétation du code JavaScript

(2 points)

La figure "*Fig.4*" (page 14) présente un extrait du code JavaScript relatif au fichier "**mesScripts.js**". En se basant sur cette figure, répondre aux questions suivantes :

- 1) Indiquer devant chaque rôle présent dans le tableau ci-dessous l'élément du langage **JavaScript** qui lui correspond.

Rôle	Élément du langage
Fonction qui convertit une valeur numérique en une chaîne.
Expression qui retourne la date courante du système.
Méthode qui retourne la valeur du jour (entre 1 et 31) d'une date donnée.
Propriété qui retourne le contenu d'un champ de saisie.
Propriété qui vérifie si une case est cochée ou non.
Méthode qui retourne une liste d'éléments ayant le même nom.
Fonction qui affiche un message dans une boîte de dialogue.
Instruction qui retourne le résultat d'une fonction.

Ne rien écrire ici

2) Soit les deux exemples ci-après de la page "Statistiques_Interventions.html".

Exemple N° 1

Gestion des Incendies

Statistiques des Incendies

Accueil Ajout des données Statistiques

Statistiques des types d'interventions

Date et heure de début d'une intervention: 01/01/2025 00:00
Date et heure de fin d'une intervention: 31/05/2025 11:59
Type(s) d'intervention(s): ☒ Aérienne ☐ Terrestre ☒ Maritime ☐ Mixte

Envoyer Annuler

Gestion des Incendies - GeoInventaire 2025

Exemple N° 2

Gestion des Incendies

Statistiques des Incendies

Accueil Ajout des données Statistiques

Statistiques des types d'interventions

Date et heure de début d'une intervention: 01/01/2025 00:00
Date et heure de fin d'une intervention: 31/05/2025 11:59
Type(s) d'intervention(s): ☐ Aérienne ☐ Terrestre ☐ Maritime ☐ Mixte

Envoyer Annuler

Gestion des Incendies - GeoInventaire 2025

a) En se basant sur le code de la fonction "inconnue" de la figure "Fig.4" (page 14) et pour chacun des deux exemples ci-dessus, donner la valeur retournée par la fonction "inconnue" tout en complétant le tableau par la trace d'exécution.

	Variable	Valeur			
Exemple N° 1	test				
	i				
	interventions[i].checked				
	La valeur retournée par la fonction "inconnue" est :				

	Variable	Valeur			
Exemple N° 2	test				
	i				
	interventions[i].checked				
	La valeur retournée par la fonction "inconnue" est :				

b) En déduire le rôle de la fonction "inconnue".

Exercice 4 : Analyse et interprétation du code PHP

(2 points)

La figure "Fig.5" (page 15) présente un extrait incomplet du code PHP relatif au fichier "Statistiques_Types.php" permettant d'afficher le nombre d'interventions pour le(s) type(s) d'intervention(s) sélectionné(s) pendant une période donnée provenant de la page "Statistiques_Interventions.html".

Pour chaque ligne de tâche manquante (ligne contenant des pointillés) de l'extrait du fichier "Statistiques_Types.php", compléter la colonne "Fonction" du tableau ci-dessous, en écrivant la fonction correspondante à chaque numéro de tâche à partir de la liste suivante :

strlen(\$ch)	mysqli_fetch_array(\$res)	mysqli_close (\$con)	count(\$types)
mysqli_connect	mysqli_query(\$con)	mysqli_connexion	substr(\$ch,0,\$L-1)
strlen (\$types)	mysqli_num_rows(\$res)	mysqli_query(\$con,\$req)	count(\$ch)

N° tâche	Fonction
1
2
3
4
5
6
7
8

EXAMEN DU BACCALAURÉAT ⚡ Session de contrôle 2025

**RÉPUBLIQUE TUNISIENNE
MINISTÈRE DE L'ÉDUCATION**

Épreuve :
**Systèmes et Technologies de
l'Informatique**

Section :
Sciences de l'informatique

Durée : **3h**

Coefficient de l'épreuve: **1.8**

N° d'inscription

--	--	--	--	--	--

GeoIncendie est un bureau d'études spécialisé dans le suivi des incendies des forêts. Pour organiser les informations nécessaires à ses statistiques, le bureau s'appuie sur une plateforme web, développée autour d'une base de données nommée "**Gestion_Incendies**". Cette base de données, dans sa version simplifiée, est représentée par le schéma textuel suivant :

Gouvernorat (codeGouv, nomGouv)

Ville (codeVille, nomVille, nbForet, codeGouv#)

Foret (codeForet, nomForet, superficieForet, codeVille#)

Incendie (codeInc, dateHeureInc, superficieBrulee, statutInc, dateHeureExt, codeForet#)

TypeIntervention (codeTypeInt, description)

Intervention (codeInt, dateHeureDebInt, dateHeureFinInt, superficielnt, codeTypeInt#, codeInc#)

Sachant que :

- **Gouvernorat** : contient les informations relatives aux gouvernorats.
- **Ville** : contient les informations relatives aux villes.
- **Foret** : regroupe les informations relatives aux forêts.
- **Incendie** : contient les informations relatives aux incendies ayant lieu dans les différentes forêts.
- **TypeIntervention** : stocke les types d'interventions utilisés pour lutter contre les incendies.
- **Intervention** : contient les informations relatives aux interventions réalisées pour maîtriser les incendies.

Soit la description des champs des tables de cette base de données :

Champ	Description	Type et propriété(s)
codeGouv	Code d'un gouvernorat	Chaîne de 3 caractères
nomGouv	Nom d'un gouvernorat	Chaîne non vide de 50 caractères et qui ne se répète pas
codeVille	Code d'une ville	Chaîne de 5 caractères
nomVille	Nom d'une ville	Chaîne non vide de 50 caractères
nbForet	Nombre de forêt dans une ville	Entier strictement positif
codeForet	Code d'une forêt	Chaîne de 6 caractères
nomForet	Nom d'une forêt	Chaîne non vide de 50 caractères
superficieForet	Superficie d'une forêt en hectares	Réel strictement positif obligatoire
codeInc	Code d'un incendie	Chaîne de 10 caractères
dateHeureInc	Date et heure de début d'un incendie	Date et Heure non vide
superficieBrulee	Superficie brûlée en hectares	Réel strictement positif obligatoire
statutInc	Statut d'un incendie	Un caractère initialisé à la lettre "A" (incendie actif). Il prend la lettre "E" si l'incendie est éteint suite à une intervention.
dateHeureExt	Date et heure d'extinction	Date et Heure (si l'incendie est actif, le champ ne contient aucune valeur)
codeTypeInt	Code d'un type d'une intervention	Chaîne de 3 caractères
description	Description du type d'une intervention	Chaîne non vide de 9 caractères
codeInt	Code d'une intervention	Chaîne de 10 caractères
dateHeureDebInt	Date et heure de début d'une intervention	Date et Heure obligatoire
dateHeureFinInt	Date et heure de fin d'une intervention	Date et Heure (si l'intervention n'est pas encore achevée, le champ ne contient aucune valeur)
superficieInt	Superficie traitée pendant une intervention, mesurée en hectares	Réel positif (sa valeur par défaut est 0)

Un aperçu de la page web "Statistiques_Interventions.html", de son code incomplet ainsi que des extraits des codes des fichiers "mesStyles.css", "mesScripts.js" et "Statistiques_Types.php" sont présentés ci-après.

❖ Page "Statistiques_Interventions.html"

- Fig.1 : Aperçu de la page "Statistiques_Interventions.html" -

```

1 <!DOCTYPE HTML>
2 <html lang="fr">
3 <head>
4 <title>Gestion des Incendies</title>
5 .....
6 <script src="mesScripts.js"></script>
7 </head>
8 .....
9 <header>
10 <h1>Gestion des Incendies</h1>
11 .....
12 <nav>
13 <div class="menu"><a href="accueil.html">Accueil</a></div>
14 <div class="menu"><a href="ajout_donnees.html#intervention">Ajout des données</a></div>
15 <div class="menu"><a href="#stat">Statistiques</a></div>
16 </nav>
17 </header>
18 <section id="#stat">
19 <h3>Statistiques des types d'interventions</h3>
20 <form id="f" action="Statistiques_Types.php" method="POST" onsubmit="return verification()">
21 <label>Date et heure de début d'une intervention:</label>
22 <input id="date1" name="date1" type="date" required>
23 <input id="temps1" name="temps1" type="time" required><br>
24 <label>Date et heure de fin d'une intervention:</label>
25 <input id="date2" name="date2" type="date" required>
26 <input id="temps2" name="temps2" type="time" required><br>
27 .....
28 <input name="intervention[]" type="checkbox" value="T11">Aérienne
29 <input name="intervention[]" type="checkbox" value="T12">Terrestre
30 <input name="intervention[]" type="checkbox" value="T13">Maritime
31 <input name="intervention[]" type="checkbox" value="T14">Mixte <br>
32 <div>
33 <input class="bouton" id="bt1" name="bt1" type="submit" value="Envoyer">
34 .....
35 </div>
36 </form>
37 </section>
38 .....
39 </body>
40 </html>

```

- Fig.2 : Code HTML de la page "Statistiques_Interventions.html" -

❖ Fichier "mesStyles.css"

<pre> 1 2 3 label { 4 display: inline-block; 5 margin: 0px 0px 5px 0px; 6 width: 300px; } 7 .bouton { 8 background-color: yellow; 9 background-size: 30px 30px; 10 width: 150px; 11 height: 30px; }</pre>	<pre> 12 .bouton:hover { text-transform: uppercase; } 13 h3, div { text-align: center; } 14 footer, header { 15 background: black; 16 text-align: center; 17 margin: 10px; 18 color: white; } 19 header { background: #007BFF; } 20 h1,h2 { margin: 5px 0px 5 px 0px; } 21 #bt1 { : url(valider.png); }</pre>
---	---

- Fig.3 : Extrait du code du fichier "mesStyles.css" -

❖ Fichier "mesScripts.js"

```

1 function date() {
2     const datej=new Date();
3     let jour=datej.getDate();
4     if (jour<10)
5         jour="0"+jour
6     let mois=datej.getMonth()+1;
7     if (mois<10)
8         mois="0"+mois
9     const annee=datej.getFullYear();
10    let datejour=String(annee)+"-"+mois+"-"+jour;
11    document.getElementById("date1").value=datejour;
12    document.getElementById("date2").value=datejour;
13 }
14 function temps() {
15     document.getElementById("temps1").value="00:00";
16     document.getElementById("temps2").value="23:59";
17 }
18 function inconnue () {
19     // sauvegarder dans la constante "interventions" les éléments HTML ayant le nom "intervention[]"
20     const interventions = document.getElementsByName("intervention[]");
21     let test = false ;
22     let i = -1;
23     do
24     {
25         i++;
26         if (interventions[i].checked == true)
27             test = true;
28     } while (test == false && i < interventions.length-1);
29     return test;
30 }
31 function verification() {
32     const date1=document.getElementById("date1").value;
33     const date2=document.getElementById("date2").value;
34     const temps1=document.getElementById("temps1").value;
35     const temps2=document.getElementById("temps2").value;
36     if (date1>date2) {
37         alert("La date de début doit être inférieure ou égale à la date de fin de l'intervention");
38         return false; }
39     if ((date1==date2) && (temps1>temps2)) {
40         alert("L'heure de début doit être inférieure ou égale à l'heure de fin de l'intervention");
41         return false; }
42     if (inconnue()==false) {
43         alert("Il faut cocher au moins un type d'intervention");
44         return false; }
45 }
```

- Fig.4 : Extrait du code du fichier "mesScripts.js" -

❖ Fichier "Statistiques_Types.php"

```

1  <?php
2  //Tâche N° 1 : Établir une connexion à la base de données.
3  $con=.....["localhost", "root", "", "gestion_incendies") or die ("Erreur de la base de données");
4  $datedebut=$_POST["date1"]." ".$_POST["temps1"];
5  $datefin=$_POST["date2"]." ".$_POST["temps2"];
6  $types=$_POST["intervention"];
7  $ch="(";
8  //Tâche N° 2 : Retourner le nombre de cases du tableau "types".
9  for($i=0; $i<.....; $i++)
10     $ch=$ch."".$types[$i].",";
11 //Tâche N° 3 : Retourner la longueur de la chaîne de caractères "ch".
12 $l=.....;
13 //Tâche N° 4 : Retourner une sous-chaîne à partir de la chaîne de caractères "ch" sans le dernier caractère.
14 $ch1=.....;
15 $ch1=$ch1."|";
16 $req="select T.description, count(I.codeInt)
17       from Intervention I, typeintervention T
18       where (I.codeTypeInt=T.codeTypeInt) and (I.codeTypeInt in $ch1) and
19             (dateHeureDebutInt>='$datedebut') and (dateHeureFinInt<='$datefin')
20       group by T.description;";
21 //Tâche N° 5 : Exécuter la requête "req".
22 $res=..... or die ("Erreur de la requête",mysql_error($con));
23 //Tâche N° 6 : Récupérer le nombre d'enregistrements retourné par la requête.
24 if (.....==0)
25     echo "Aucune intervention n'est enregistrée pour les critères introduits.";
26 else
27     echo "<table border=1><thead><tr><th>Type Interventions</th><th>Nombre Interventions</th></tr> </thead><tbody>";
28 //Tâche N° 7 : Retourner un enregistrement du résultat de l'exécution de la requête.
29 while ($enreg=.....)
30     echo "<tr><td>$enreg[0]</td>
31           <td>$enreg[1]</td></tr>";
32 echo "</tbody></table>";
33 //Tâche N° 8 : Se déconnecter de la base de données.
34 .....;
35 ?>

```

- Fig.5 : Extrait du code du fichier "Statistiques_Types.php" -