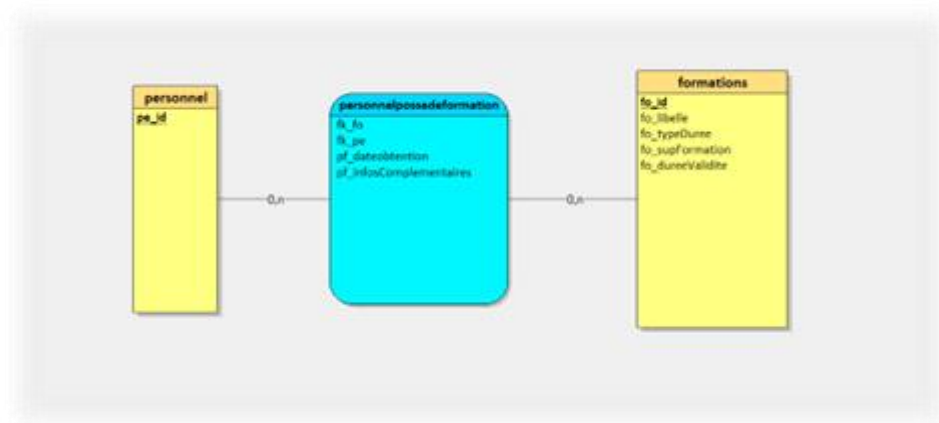


FORMATION

Insérer une base de données pour un site internet de WOIT-GROUP et voir les formations du personnel de l'entreprise en le montrant sur le site. Je dois utiliser un logiciel qui se nomme PGADMIN pour pouvoir effectuer ceci. J'ai un modèle qui peut m'aider et un script SQL pour pouvoir faire ceci :



```
CREATE TABLE formations (
  fo_id integer NOT NULL,
  fo_libelle character varying(255),
  fo_dureeValidite integer,
  fo_typeDuree character varying(255),
  fo_supFormation boolean DEFAULT false
);
ALTER TABLE formations OWNER TO postgres;

CREATE SEQUENCE formations_fo_id_seq
  AS integer
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;

ALTER TABLE public.formations_fo_id_seq OWNER TO postgres;
ALTER TABLE ONLY formations ALTER COLUMN fo_id SET DEFAULT nextval('formations_fo_id_seq'::regclass);

CREATE TABLE personnelpossedeformation (
  fk_pe integer NOT NULL,
  fk_fo integer NOT NULL,
  pf_dateobtention timestamp without time zone,
  pf_infosComplementaires character varying(255)
);
ALTER TABLE ONLY personnelpossedeformation
  ADD CONSTRAINT personnelpossedeformation_pkey PRIMARY KEY (fk_fo, fk_pe);
```

J'ai dû mettre cela dans la base de données sur pgAdmin. Puis j'ai attaqué le code. Voici le code effectué :

- Fonctions de calcul de date :

```

        if ($personnel == null) {
            $personnel = $bdd->getNextPersonnelByPrenomAndNom($p['pe_prenom'], $p['pe_nom']);
        }
    }

    print_r($listeFormation);

    $p = $bdd->getPersonnelByPE_id($GET['id']);
    if (isset($GET['type'])) {
        if ($GET['type'] == "next") {
            $personnel = $bdd->getNextPersonnelByPrenomAndNom($p['pe_prenom'], $p['pe_nom']);
            if ($personnel == null) {
                $personnel = $p;
            }
        } else {
            $personnel = $bdd->getPreviousPersonnelByPrenomAndNom($p['pe_prenom'], $p['pe_nom']);
            if ($personnel == null) {
                $personnel = $p;
            }
        }
    } else {
        $personnel = $p;
    }

    references:
    function calculerDateFin($dateDebut, $duree, $typeDuree) {
        $dateDeb = date_create($dateDebut);

        if ($typeDuree == "aaa") {
            $numAnnee = (int)($dateDeb->format("Y")) + (int)$duree;
            $dateDeb = date_create($dateDeb->format("$numAnnee.-m.-d"));
        }
        return $dateDeb->format("d-m-y");
    }

    references:
    function calculerDateFinMois($dateDebut, $duree, $typeDuree) {
        $dateDebM = date_create($dateDebut);

        if ($typeDuree == "aaa") {
            $numAnnee = (int)($dateDebM->format("Y")) + (int)$duree;
            $dateDebM = date_create($numAnnee . $dateDebM->format("-m-d"));
        }
        elseif ($typeDuree == "mm") {
            $numMois = (int)($dateDebM->format("m")) + (int)$duree;
            $numAnnee = (int)($dateDebM->format("Y"));
            while ($numMois > 12) {
                $numMois -= 12;
                $numAnnee++;
            }
            $dateDebM = date_create($numAnnee . '-' . $numMois . $dateDebM->format("-d"));
        }

        $dateFinM = date_create();
        $diff = date_diff($dateDebM, $dateFinM);
    }

```

```

    $dateFinM = date_create();
    $diff = date_diff($dateDebM, $dateFinM);

    $nbMois = $diff->format("%m") + 12 * $diff->format("%y");

    return $nbMois;

```

- Affichage sur la page web :

```

<?php
if(isset($ListeFormation[0])){
    echo "<label for=\"Formation\">Formation</label>";
    echo "<table class=\"table-formationnv1\">";
    if (is_array($ListeFormation[0])) {
        echo "<tr>";
        echo "<td>Nom formation :</td>";
        echo "<td>Durée de validité :</td>";
        echo "<td>Date obtention :</td>";
        echo "<td>Date de fin de validité :</td>";
        echo "</tr>";
        foreach ($ListeFormation as $formation) {
            echo "<tr>";
            echo "<td>" . $formation['fo_libelle'] . "</td>";
            echo "<td>" . $formation['fo_dureevalidite'] . " ans</td>";
            echo "<td>" . formatDate($formation['pf_dateobtention']) . "</td>";
            echo "<td>" . calculerDateFin($formation['pf_dateobtention'], $formation['fo_dureevalidite']) . "</td>";
            echo "</tr>";
        }
    } else {
        echo "<tr>";
        echo "<td>Nom formation :</td>";
        echo "<td>Durée de validité :</td>";
        echo "<td>Date obtention :</td>";
        echo "<td>Date de fin de validité :</td>";
        echo "</tr>";
        echo "<tr>";
        echo "<td>" . $ListeFormation['fo_libelle'] . "</td>";
        echo "<td>" . $ListeFormation['fo_dureevalidite'] . " ans</td>";
        echo "<td>" . formatDate($ListeFormation['pf_dateobtention']) . "</td>";
        echo "<td>" . calculerDateFin($ListeFormation['pf_dateobtention'], $ListeFormation['fo_dureevalidite']) . "</td>";
        echo "</tr>";
    }
}
?>

```

- Ici on récupère les modifications :

```
1 reference | 0 overrides
function getFormationById($FO_id){
    $reqSql = $this->conn->prepare(
        "SELECT * FROM formations
        LEFT JOIN personnelposse deformation ON fk_fo = fo_id
        WHERE fo_supformation = false
        AND fk_PE = :PE_id"
    );
    $reqSql->bindValue(':PE_id', $this->secure($FO_id));
    $reqSql->execute();
    return $reqSql->fetch(); }
}
```

- CSS :

```

86 .table-formationnvl {
87     font-family: Verdana;
88     border: 1px solid #rgb(138, 138, 138);
89     width: 100%;
90     background-color: #ddede1;
91     border-radius: 5px;
92     margin: auto;
93     text-align: center;
94     border-spacing: 0;
95 }
96
97 .table-formationnvl td {
98     position: relative;
99     padding-bottom: 10px;
100    padding : 10px;
101    padding-right: 15px;
102 }
103
104 .table-formationnvl td::after {
105     content: "";
106     position: absolute;
107     left: 0;
108     bottom: -2px;
109     width: 100%;
110     height: 1px;
111     background-color: #rgb(138, 138, 138);
112 }
113
114 .table-formationnvl tr:last-child td::after {
115     display: none;
116 }

```

J'ai mis des valeurs test pour essayer et le code effectue tous les changements automatiques en termes de données.

J'ai aussi mis en place un tri de données par la date de validité la plus proche à la plus lointaine. Pour cela j'ai ajouté un ORDER BY :

```

1 reference | 0 overrides
function getFormationById($FO_id){
    $reqSql = $this->conn->prepare(
        "SELECT * FROM formations
        LEFT JOIN personnepossedeformation ON fk_fo = fo_id
        WHERE fo_supformation = false
        AND fk_PE = 'PE_id
        ORDER BY fo_dureevalidite ASC"
    );
    $reqSql->bindValue(':PE_id', $this->secure($FO_id));
    $reqSql->execute();
    return $reqSql->fetchAll(); }
}

```

Voici le résultat :

Formation

Nom formation :	Durée de validité :	Date obtention :	Date de fin de validité :
Aptitude médical	2 ans	26-05-23	26-05-25
ATEX	3 ans	26-05-23	26-05-26
Formation sécurité N2	4 ans	26-05-23	26-05-27
Travail en hauteur et port du harnais	5 ans	26-05-23	26-05-28