Créer un trigger qui va empêcher de saisir un étudiant ayant âge <17

|  |
| --- |
| create TRIGGER tr\_Ajout\_age BEFORE INSERT on etudiant for each ROW  BEGIN  declare x int;  set x= new.ageE;  if x<17 THEN  SIGNAL SQLSTATE '45000'  SET MESSAGE\_TEXT=" L'age doit être >17,";  end if;  end |
| create TRIGGER tr\_Ajout\_age BEFORE INSERT on etudiant for each ROW  BEGIN  if new.ageE<17 THEN  SIGNAL SQLSTATE '45000'  SET MESSAGE\_TEXT=" L'age doit être >17,";  end if;  end |

Créer un trigger qui empêche de modifier l’âge de l’étudiant avec une valeur <16

|  |
| --- |
| [create](http://127.0.0.1/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/create-trigger.html" \t "mysql_doc) [TRIGGER](http://127.0.0.1/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/create-trigger.html" \t "mysql_doc) tr\_Modif\_age before [update](http://127.0.0.1/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/update.html" \t "mysql_doc) on etudiant for each ROW BEGIN [declare](http://127.0.0.1/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/declare.html" \l "declare" \t "mysql_doc) x [int](http://127.0.0.1/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/numeric-types.html" \t "mysql_doc); [set](http://127.0.0.1/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/set.html" \t "mysql_doc) x= new.ageE; [if](http://127.0.0.1/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/control-flow-functions.html" \l "function_if" \t "mysql_doc) x<17 THEN SIGNAL SQLSTATE '45000' [SET](http://127.0.0.1/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/set.html" \t "mysql_doc) MESSAGE\_TEXT="Problème de modification: L'age doit être >17"; end [if](http://127.0.0.1/phpmyadmin/url.php?url=http://dev.mysql.com/doc/refman/5.5/en/control-flow-functions.html" \l "function_if" \t "mysql_doc); end |

Créer un trigger qui va empêcher d’ajouter un étudiant avec le nom moins de 3 caractères.

|  |
| --- |
| CREATE TRIGGER `tr\_Ajout\_Nom` BEFORE INSERT ON `etudiant`  FOR EACH ROW BEGIN  if length( new.nomE)<3 THEN  SIGNAL SQLSTATE '45000'  set MESSAGE\_TEXT='Le nom doit comporter plus de 3 caratères';  end if;  end |

Créer un trigger qui va empêcher d’ajouter un étudiant avec un email déjà saisi pour un autre étudiant.

|  |
| --- |
| CREATE TRIGGER `tr\_ajout\_Email` BEFORE INSERT ON `etudiant`  FOR EACH ROW BEGIN  DECLARE x varchar(50);  set x= new.emailE;  if EXISTS( select \* from etudiant WHERE emaile= x) THEN  SIGNAL SQLSTATE '45000'  set MESSAGE\_TEXT="Cet email existe déjà";  end if;  end |

Créer un trigger qui va empêcher de modifier l’email d’un étudiant.

|  |
| --- |
| CREATE TRIGGER `tr\_ModifEmail2` BEFORE UPDATE ON `etudiant`  FOR EACH ROW begin  declare oldemail, newemail varchar(50);  set oldemail=old.emailE;  set newemail=new.emailE;  if newemail != oldemail then  signal sqlstate '45000' set message\_Text="On ne peut pas modifier l'email";  end if;  end |

Créer un trigger qui va empêcher de supprimer un étudiant qui a passé des examens.

Créer un trigger qui augment l’effectif de la classe chaque fois on ajout un nouvel étudiant.

Créer un trigger qui décrémenter l’effectif de la classe chaque fois on supprime un nouvel étudiant.

Créer un trigger qui va mettre à jour l’effectif des classe lorsqu’un étudiant change de classe, on décrémente l’effectif de la classe d’origine et on incrémente celui de la classe de destination.

Créer un trigger qui empêche d’inscrire un étudiant dans une classe dont l’effectif dépasse 25 étudiants.

Créer un trigger qui va empêcher de saisir une note dont le type n’est pas ‘CC’ ou ‘EFM’.

Créer un trigger qui va empêcher de saisir une note >20 pour une notation de type ‘CC’ , et une note >40 pour une notation de type ‘EFM’.