```
CREATE DEFINER=`root`@`localhost` FUNCTION `GETSALESREP`(vCOUNTRY VARCHAR(50)) RETURNS int(11)
BEGIN
     DECLARE VSALESREP INT(11);
     SET vSALESREP = ( SELECT SALESREPEMPLOYEENUMBER
                         FROM CUSTOMERS
                        WHERE SALESREPEMPLOYEENUMBER IN ( SELECT EMPLOYEENUMBER
                                                           FROM EMPLOYEES
                                                           WHERE JOBTITLE = 'SALES REP'
                                                            AND OFFICECODE IN (SELECT OFFICECODE
                                                                                 FROM OFFICES
                                                                                WHERE COUNTRY = vCOUNTRY
                     GROUP BY SALESREPEMPLOYEENUMBER
                     ORDER BY COUNT (customernumber) ASC
                        LIMIT 1
                      );
     RETURN vSALESREP;
END
CREATE DEFINER=`root`@`localhost` FUNCTION `GETRECEIVABLE` (VCUSTOMERNUMBER INT(11), VSTARTDATE DATE, VENDDATE DATE) RETURNS double
BEGIN
 DECLARE VRECEIVABLE DOUBLE;
 SET VRECEIVABLE = (SELECT Y.ORDERS-X.PAYMENTS
                       FROM ( SELECT C.CUSTOMERNUMBER, SUM(AMOUNT) AS PAYMENTS
                               FROM CUSTOMERS C JOIN PAYMENTS P ON C.CUSTOMERNUMBER=P.CUSTOMERNUMBER
                              WHERE C.CUSTOMERNUMBER = VCUSTOMERNUMBER
                                AND P.PAYMENTDATE <= VENDDATE
                                AND P.PAYMENTDATE >= VSTARTDATE
                           GROUP BY C.CUSTOMERNUMBER) X NATURAL JOIN ( SELECT O.CUSTOMERNUMBER, SUM(OD.QUANTITYORDERED*OD.PRICEEACH)
AS ORDERS
                                                                          FROM ORDERS O JOIN ORDERDETAILS OD ON
O.ORDERNUMBER=OD.ORDERNUMBER
                                                                          WHERE O.CUSTOMERNUMBER = VCUSTOMERNUMBER
                                                                           AND O.ORDERDATE <= VENDDATE
                                                                           AND O.ORDERDATE >= VSTARTDATE
                                                                      GROUP BY O.CUSTOMERNUMBER) Y
                             );
 RETURN VRECEIVABLE;
END
CREATE DEFINER=`root`@`localhost` FUNCTION `GETORDERS` (VCUSTOMERNO INT(11),
                                                        VSTARTDATE DATE,
                                                         VENDDATE DATE) RETURNS double
BEGIN
     DECLARE VORDERS DOUBLE;
     SET VORDERS = (SELECT SUM(OD.PRICEEACH * OD.QUANTITY)
                      FROM ORDERDETAILS OD JOIN ORDERS O
                                               ON OD.ORDERNUMBER=O.ORDERNUMBER
                     WHERE O.ORDERDATE >= VSTARTDATE
                             O.ORDERDATE <= VENDDATE
                     AND
```

```
AND O.CUSTOMERNUMBER = VCUSTOMERNO
                  GROUP BY O.CUSTOMERNUMBER
                  );
     RETURN VORDERS;
END
CREATE DEFINER=`root`@`localhost` TRIGGER `dbsales`.`customers BEFORE INSERT` BEFORE INSERT ON `customers` FOR EACH ROW
     IF (NEW.SALESREPEMPLOYEENUMBER IS NULL) THEN
          SET NEW.SALESREPEMPLOYEENUMBER = GETSALESREP(NEW.COUNTRY);
     SET NEW.CREDITLIMIT = 0;
END
CREATE DEFINER=`root`@`localhost` TRIGGER `dbsales`.`employees_AFTER UPDATE` AFTER UPDATE ON `employees` FOR EACH ROW
     IF OLD.JOBTITLE = 'SALES REP' THEN
          UPDATE CUSTOMERS
          SET SALESREPEMPLOYEENUMBER = GETSALESREP (COUNTRY)
           WHERE SALESREPEMPLOYEENUMBER = OLD.EMPLOYEENUMBER;
   END IF;
END
```