

E531**BEGIN** ≡ PROCEDURE sp_fibonacci (IN nombre INT)

DECLARE v_nmoins2 INT DEFAULT 0;

DECLARE v_nmoins1 INT DEFAULT 1;

DECLARE v_n INT DEFAULT 1;

DECLARE i INT DEFAULT 0;

DECLARE flagNotFound BOOLEAN DEFAULT false;**BEGIN****DECLARE EXIT HANDLER FOR 1051 SET flagNotFound := true;**

DROP TABLE TMP_FIBO;

END;

CREATE TEMPORARY TABLE TMP_FIBO (indice INT, valeur INT);

INSERT INTO TMP_FIBO VALUES (i, 1);

WHILE i<nombre DO

SET i:=i+1;

SET v_n:=v_nmoins1 + v_nmoins2;

INSERT INTO TMP_FIBO VALUES (i, v_n);

SET v_nmoins2 := v_nmoins1;

SET v_nmoins1 := v_n;

END WHILE;

SELECT 'Fin de la création.' AS 'résultat';

END

```

E532 CREATE PROCEDURE sp_liste_abonne_annuel ( )
BEGIN
    DECLARE v_aboid INT;
    DECLARE v_annuel BOOLEAN;
    DECLARE v_ident varchar(40);

    DECLARE flagNotFound BOOLEAN DEFAULT false;
    DECLARE      fincurs1 BOOLEAN DEFAULT 0;
    /* ATTENTION toutes les déclarations de variables
    doivent être placées avant la définition d'un curseur */

    DECLARE curs1 CURSOR FOR
        SELECT      DISTINCT abo_id, CONCAT(abo_nom,' ',abo_prn)
        FROM        ABONNE;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET fincurs1:=1;

    BEGIN
        DECLARE EXIT HANDLER FOR 1051 SET flagNotFound := true;
        DROP TABLE TMP_ABOAN;
    END;
    CREATE TEMPORARY TABLE TMP_ABOAN (numéro INT, identité VARCHAR(40));

    OPEN curs1;
    FETCH curs1 INTO v_aboid, v_ident;
    WHILE NOT fincurs1 DO
        CALL  sp_abonne_annuel(v_aboid,v_annuel);
        IF v_annuel
        THEN
            INSERT INTO TMP_ABOAN VALUES (v_aboid, v_ident);
        END IF;
        FETCH curs1 INTO v_aboid, v_ident;
    END WHILE;
    CLOSE curs1;
END

```

