

Type booléen	Type caractère	Type chaîne de caractères																					
<p>❁ <u>Domaine de valeur</u> :</p> <p>Deux valeurs logiques <b>VRAI</b> (TRUE) et <b>FAUX</b> (FALSE)</p> <p>❁ <u>Opérateurs logiques sur les booléen</u> :</p> <p>.....</p> <p>.....</p> <p>❁ <u>Tableau de vérité</u></p> <p>❁ <u>Déclaration d'une variable booléenne</u> :</p> <p>☒ <u>Analyse/Algorithme</u> :</p> <p style="text-align: center;">TDO</p> <table border="1" data-bbox="123 1093 580 1169"> <thead> <tr> <th>Objet</th> <th>Nature/Type</th> <th>Rôle</th> </tr> </thead> <tbody> <tr> <td>Nom_var</td> <td>Var/BOOLEEN</td> <td>Rôle</td> </tr> </tbody> </table> <p>☒ <u>Turbo Pascal</u> :</p> <p>VAR Nom_var : BOOLEAN ;</p>	Objet	Nature/Type	Rôle	Nom_var	Var/BOOLEEN	Rôle	<p>❁ <u>Domaine de valeur</u> :</p> <p><i>Exemple</i> : "A" ('A'), "3" ('3')</p> <p>☑ Les caractères alphanumériques imprimables comme les lettres (minuscule et majuscule), les chiffres et les symboles</p> <p>☑ Les caractères spéciaux non imprimables comme le retour chariot, Echap ...</p> <p>On peut comparer deux caractères selon leurs codes ASCII</p> <p><u>Caractère</u> : ! &lt; 0 &lt; 9 &lt; A &lt; Z &lt; a &lt; z</p> <p><u>Code ASCII</u> : 33 &lt; 48 &lt; 57 &lt; 65 &lt; 90 &lt; 97 &lt; 122</p> <p>❁ <u>La concaténation des caractères</u> :</p> <p><i>Exemple</i> : "a"+"B" = "aB", "1"+"8" = "18".</p> <p>❁ <u>Les opérateurs relationnels</u></p> <p><i>Exemple</i> : "A" &lt; "B" est Vrai car 65 &lt; 66</p> <p>❁ <u>Déclaration</u> :</p> <p>☒ <u>Analyse/Algorithme</u> :</p> <p style="text-align: center;">TDO</p> <table border="1" data-bbox="705 1121 1202 1198"> <thead> <tr> <th>Objet</th> <th>Nature/Type</th> <th>Rôle</th> </tr> </thead> <tbody> <tr> <td>Nom_var</td> <td>Var/CARACTERE</td> <td>Rôle</td> </tr> </tbody> </table> <p>☒ <u>Turbo Pascal</u> :</p> <p>VAR Nom_var : CHAR ;</p> <p>❁ <u>Les fonctions prédéfinies</u></p>	Objet	Nature/Type	Rôle	Nom_var	Var/CARACTERE	Rôle	<p>❁ <u>Définition</u> : Une suite de n caractères avec n compris entre 0 et 255. Si n=0 on dit que la chaîne est vide.</p> <p><u>Exemple</u> :</p> <p>"Informatique" 'Informatique'</p> <p>❁ <u>Les opérateurs sur les chaînes</u> :</p> <p>☐ "ALI" + " BABA" = "ALI BABA "</p> <p>☐ On peut utiliser les opérateurs Relationnels sur les chaînes:</p> <p><u>Exemple</u> : "Mohamed" &gt; "Ali"</p> <p>"Said" &lt; "Saleh"</p> <p><u>Remarque</u>: Comparaison caractère par caractère.</p> <p>❁ <u>L'accès à un caractère d'une chaîne</u> :</p> <p>On peut accéder au ième caractère d'une chaîne CH en utilisant la notion CH[i] où 1 ≤ i ≤ long(ch).</p> <p><u>Exemple</u> :</p> <p>CH ← "Tunisie" c ← ch[1]</p> <p>ch[7] ← "a" donc ch devient "Tunisia"</p> <p>❁ <u>Déclaration</u> :</p> <p>☒ <u>Analyse/Algorithme</u> : TDO</p> <table border="1" data-bbox="1361 1042 1951 1182"> <thead> <tr> <th>Objet</th> <th>Nature/Type</th> <th>Rôle</th> </tr> </thead> <tbody> <tr> <td>Nom_var</td> <td>Var/CHAINE</td> <td>Rôle</td> </tr> <tr> <td>Nom_var</td> <td>Var/CHAINE[taille_max]</td> <td>Rôle</td> </tr> </tbody> </table> <p>☒ <u>Turbo Pascal</u> :</p> <p>VAR Nom_var : STRING ;</p> <p>Nom_var : STRING[taille_max] ;</p> <p>❁ <u>Les fonctions prédéfinies</u></p>	Objet	Nature/Type	Rôle	Nom_var	Var/CHAINE	Rôle	Nom_var	Var/CHAINE[taille_max]	Rôle
Objet	Nature/Type	Rôle																					
Nom_var	Var/BOOLEEN	Rôle																					
Objet	Nature/Type	Rôle																					
Nom_var	Var/CARACTERE	Rôle																					
Objet	Nature/Type	Rôle																					
Nom_var	Var/CHAINE	Rôle																					
Nom_var	Var/CHAINE[taille_max]	Rôle																					