Nο	SubSig boolean isEmpty()	APIPair		RegexWrapper				
No.		API Invocation	CondExpr	R	L	S	Lmin	Relation
1		caller: str array r=str.isEmpty()	r==T	[\s\S]{0}	-1	NULL	0	<u> </u>
	boolean equals(Object)	r=str.equals(v)	r==T	V	-1	NULL	Len(v)	=
	int compareTo(String)	r=str.compareTo(v)	r==0	V	-1	NULL	Len(v)	≡
	int compareTo(Object)	r=str.compareTo(v) r=str.contentEquals(v)	r==0 r==T	V	-1 -1	NULL NULL	Len(v)	=
	boolean contentEquals(CharSequence) boolean contentEquals(StringBuffer)	r=str.contentEquals(v) r=str.contentEquals(v)	r==1 r==T	V V	-1 -1	NULL	Len(v)	=
7	int compareToIgnoreCase(String)	r=str.compareToIgnoreCase(v)	r==0	V	-1	NULL	Len(v)	⊈
-	boolean equalsIgnoreCase(String)	r=str.equalsIgnoreCase(v)	r==T	V	-1	NULL	Len(v)	⊈
	boolean contains(CharSequence) boolean startsWith(String)	r=str.contains(v) r=str.startsWith(v)	r==T r==T	v[\s\S]* v[\s\S]*	-1 -1	NULL NULL	Len(v)	⊈
	boolean startsWith(String,int)	$r=str.startsWith(v,i_1)$	r==T	$[\slashed] = [\slashed] = [\s$	-1	NULL	i_1 +Len(v)	
-	boolean endsWith(String)	r=str.endsWith(v)	r==T	[\s\S]*v	-1	NULL	Len(v)	≡
		r=str.indexOf(v)	$r==i_1$	$[^{v}]\{i_{1}\}v[^{s}]^{*}$	-1	NULL	i_1 +Len(v)	
13	int indexOf(String)	r=str.indexOf(v)	r>=i ₁	[^v]{i ₁ ,}v[\s\S]*	-1	NULL	i_1 +Len(v)	
		r=str.indexOf(v)	r<=i ₁	$[^{v}]\{0,i_{1}\}v[\s\s]^{*}$	-1	NULL	Len(v)	⊈ _
		r=str.indexOf(v) r=str.indexOf(v)	$r==i_1$ $r>=i_1$	$[^v]\{i_1\}v[^s]^*$ $[^v]\{i_1\}v[^s]^*$	-1 -1	NULL NULL	i_1 +Len(v) i_1 +Len(v)	
14	int indexOf(int)	r=str.indexOf(v)	$r <= i_1$	$[v]\{0,i_1\}v[s]$	-1	NULL	Len(v)	=
		()	1	(convert v to string first)			()	
	int indexOf(String,int) int indexOf(int,int)	$r=str.indexOf(v,i_1)$	$r==i_2$	$[\s\s]\{i_1\}[\v]\{i_2-i_1\}\v[\s\s]*$	-1	NULL	i ₂ +Len(v)	
15		r=str.indexOf(v,i ₁)	r>=i ₂	$[\s\] \{i_1\} [\v] \{i_2-i_1,\} v[\s\]^*$	-1	NULL	i ₂ +Len(v)	<u>† </u>
		r=str.indexOf(v , i_1)	r<=i ₂	$[\s\s] \{i_1\} [\v] \{0, i_2 - i_1\} v [\s\s] *$	-1		i_1 +Len(v)	•
		r=str.indexOf(v,i ₁) r=str.indexOf(v,i ₁)	$r==i_2$ $r>=i_2$	$[\s\S]\{i_1\}[\v]\{i_2-i_1\}\v[\s\S]^*$ $[\s\S]\{i_1\}[\v]\{i_2-i_1,\}\v[\s\S]^*$	-1 -1	NULL NULL	i_2 +Len(v) i_2 +Len(v)	1
16		r=str.indexOf(v , i_1)	$r <= i_2$		-1 -1	NULL	i_1 +Len(v)	
		1 sumacrof(v,il)	1 12	(convert v to string first)	1	IVELL	I ₁ · Len(v)	
	int lastIndexOf(String)	r=str.lastIndexOf(v)	$r==i_1$	$[\s\][i] v[^v]^*$	-1	NULL	i ₁ +Len(v)	⊈
17		r=str.lastIndexOf(v)	r>=i ₁	$[\s\S]\{i_1,\}v[\v]^*$	-1	NULL	i ₁ +Len(v)	
		r=str.lastIndexOf(v)	r<=i1	$[\s\s] \{0,i_1\} v [\v]^*$	-1	NULL	Len(v)	⊈
	int lastIndexOf(int) int lastIndexOf(String,int)	r=str.lastIndexOf(v)	r==i ₁	[\s\S]{i ₁ }v[^v]*	-1	NULL	i_1 +Len(v)	
18		r=str.lastIndexOf(v) r=str.lastIndexOf(v)	r>=i ₁	[\s\S]{i ₁ ,}v[^v]* [\s\S]{0 i.}v[^v]*	-1 -1	NULL NULL	i_1 +Len(v)	=
		1-5u.lasumucxOI(V)	r<=i ₁	$[\s\s] \{0,i_1\} v [\v]^*$ (convert v to string first)	-1	NULL	Len(v)	_
		r=str.lastIndexOf(v,i ₁)	$r==i_2$	$[\s\s]\{i_2\}\ v[\v]\{i_1+1-i_2-Len(v)\}[\s\s]^*$	-1	NULL	i ₁ +1	⊈
19		$r=str.lastIndexOf(v,i_1)$	$r>=i_2$		-1	NULL	i ₁ +1	⊈
		r=str.lastIndexOf(v , i_1)	$r \le i_2$	$ [\s\s] \{0,i_2\} v [\v] \{i_1+1-i_2-Len(v)\} [\s\s]^* $	-1	NULL	i ₁ -i ₂ +1	⊈
		r =str.lastIndexOf(v, i_1)	$r==i_2$	$[\s\s]\{i_2\}\ v[\v]\{i_1+1-i_2-Len(v)\}[\s\s]^*$	-1	NULL	i ₁ +1	=
20	int lastIndexOf(int,int)	$r=str.lastIndexOf(v,i_1)$	r>=i ₂	$[\s\s]\{i_2,\sv[\v]\{i_1+1-i_2-Len(v)\}[\s\s]*$		NULL	i ₁ +1	⊈
21		$r=str.lastIndexOf(v,i_1)$	r<=i ₂	$[\s\s] \{0,i_2\} v [\v] \{i_1+1-i_2-Len(v)\} [\s\s]^*$		NULL	$i_1 - i_2 + 1$	⊈
	boolean matches(String)	r=str.matches(v)	r==T	V [\a\C](;) and Ctuf\a\C]*	-1	NULL	$L_{\min}(v)$	=
77	boolean regionMatches (int,String,int,int)	$r=str.regionMatches(i_1,v_1,i_2,i_3)$	r==T		-1	NULL	$i_1 + i_3$	=
+	boolean regionMatches			$[\s\S]\{i_1\} subStr[\s\S]^*$				
73	(boolean,int,String,int,int)	r=str.regionMatches(b_1 , i_1 , v_1 , i_2 , i_3)	r==T	$subStr=v_1.substring(i_2,i_2+i_3);$	-1	NULL	$i_1 + i_3$	⊈
		$r=str.charAt(i_1)$	$r==i_2$	$[\s\s]\{i_1\}i_2[\s\s]^*$	-1	NULL	i_1+1	=
24	char charAt(int)	$r=str.charAt(i_1)$	$r>=i_2$	$[\s\][i_1\}[i_2+1][\s\]^*$	-1	NULL	i_1+1	⊈
		$r=str.charAt(i_1)$	$r <= i_2$	$[\s\]\{i_1\}[i_2-1][\s\]^*$	-1	NULL	i_1+1	⊈
		r=str.codePointAt(i ₁)	$r==i_2$	$[\s\S]\{i_1\}i_2[\s\S]^*$	-1	NULL	i ₁ +1	=
25	int codePointAt(int)	$r=str.codePointAt(i_1)$	$r>=i_2$	$[\s\S]\{i_1\}[i_2+1][\s\S]^*$	-1	NULL	i ₁ +1	⊈
		r=str.codePointAt(i ₁)	r<=i2	[\s\S]{i ₁ }[i ₂ -1][\s\S]*	-1	NULL	i ₁ +1	⊈
26	int codePointCount(int,int) int codePointBefore(int)	r=str.codePointCount(i ₁ ,i ₂)	r==i ₃	[\s\S]{i ₂ ,}	-l	NULL	i ₂	≡
20		r=str.codePointCount(i ₁ ,i ₂) r=str.codePointCount(i ₁ ,i ₂)	$r>=i_3$ $r<=i_3$		-1 _1	NULL NULL	i_2 i_2	⊈
		r=str.codePointBefore(i ₁)	$r==i_2$	$[\s\s]\{i_1-1\}i_2[\s\s]^*$	-1	NULL	i_1	≠ ≡
27		r=str.codePointBefore(i ₁)	$r>=i_2$	$[\s\][i_1-1][i_2+1][\s\]^*$	-1	NULL	\mathbf{i}_1	⊈
27		r=str.codePointBefore(i ₁)	$r <= i_2$	[\s\S]{i ₁ -1}[i ₂ -1][\s\S]*	-1	NULL	i_1	⊈
				(convert i_2 -1, i_2 +1, i_2 to string first)				
			$r==i_3$					
28	int offsetBvCodePoints(int.int)	r=str.offsetBvCodePoints(i,.i ₂)	113	$\lceil s \rceil \{i_1+i_2\}$	-1	NULL	i_1+i_2	=
28	int offsetByCodePoints(int,int)	r=str.offsetByCodePoints(i ₁ ,i ₂)	(>=,<=,>,<)	$[\s\]\{i_1+i_2,\}$	-1		i_1+i_2	=
		r=str.length()	(>=,<=,>,<) r==i ₁	$[\s\S]\{i_1\}$	-1 -1	NULL	i ₁	=
	int offsetByCodePoints(int,int) int length()	r=str.length() r=str.length()	$(>=,<=,>,<)$ $r==i_1$ $r>=i_1$			NULL NULL	$egin{array}{c} egin{array}{c} \egin{array}{c} \egin{array}{c} \egin{array}{c} \egin{array}{c} \egin{array}{c} \egin{array}$	
		r=str.length()	(>=,<=,>,<) r==i ₁	$[\s\S]\{i_1\}$		NULL	i ₁	=
29	int length()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray()	(>=,<=,>,<) r==i ₁ r>=i ₁ r<=i ₁ r(i)==v r[i]>v	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{0,i ₁ } [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]*	-1 -1 -1 -1 -1	NULL NULL NULL NULL NULL	i_1 i_1 0 $i+1$ $i+1$	= = = = ≠
29		r=str.length() r=str.length() r=str.length() r=str.toCharArray()	$(>=,<=,>,<)$ $r==i_1$ $r>=i_1$ $r<=i_1$ $r(i)==v$	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{0,i ₁ } [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]*	-1 -1 -1 -1	NULL NULL NULL	i_1 i_1 0 $i+1$	= = = =
29	int length()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray()	(>=,<=,>,<) r==i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i] <v< td=""><td>[\s\S]{i₁} [\s\S]{i₁,} [\s\S]{0,i₁} [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first)</td><td>-1 -1 -1 -1 -1 -1</td><td>NULL NULL NULL NULL NULL NULL</td><td>i_1 i_1 0 $i+1$ $i+1$ $i+1$</td><td>= = = = ≠</td></v<>	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{0,i ₁ } [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first)	-1 -1 -1 -1 -1 -1	NULL NULL NULL NULL NULL NULL	i_1 i_1 0 $i+1$ $i+1$ $i+1$	= = = = ≠
30	int length() char[] toCharArray()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray()	(>=,<=,>,<) r==i ₁ r>=i ₁ r<=i ₁ r(i)==v r[i]>v r[i] <v r[i]="=v</td"><td>[\s\S]{i₁} [\s\S]{i₁,} [\s\S]{0,i₁} [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}v[\s\S]*</td><td>-1 -1 -1 -1 -1</td><td>NULL NULL NULL NULL NULL</td><td>i_1 i_1 0 $i+1$ $i+1$</td><td>= = = = ≠ ⊈</td></v>	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{0,i ₁ } [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}v[\s\S]*	-1 -1 -1 -1 -1	NULL NULL NULL NULL NULL	i_1 i_1 0 $i+1$ $i+1$	= = = = ≠ ⊈
30	int length()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray()	(>=,<=,>,<) r==i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i] <v< td=""><td>[\s\S]{i₁} [\s\S]{i₁,} [\s\S]{0,i₁} [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first)</td><td>-1 -1 -1 -1 -1 -1</td><td>NULL NULL NULL NULL NULL NULL NULL</td><td>i₁ i₁ 0 i+1 i+1 i+1 i+1</td><td>= = = = ⊈ ⊈</td></v<>	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{0,i ₁ } [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first)	-1 -1 -1 -1 -1 -1	NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1 i+1 i+1 i+1	= = = = ⊈ ⊈
30	int length() char[] toCharArray()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes(v ₁) r=str.getBytes(v ₁)	(>=,<=,>,<) r==i ₁ r>=i ₁ r<=i ₁ r(i)==v r[i]>v r[i] <v r[i]="">v r[i]<v< td=""><td></td><td>-1 -1 -1 -1 -1 -1 -1</td><td>NULL NULL NULL NULL NULL NULL NULL NULL</td><td>i₁ i₁ 0 i+1 i+1 i+1 i+1 i+1 i+1</td><td>= = = = = = = = = = = = = = = = = = =</td></v<></v>		-1 -1 -1 -1 -1 -1 -1	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1 i+1 i+1 i+1 i+1 i+1	= = = = = = = = = = = = = = = = = = =
30	int length() char[] toCharArray()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes(v ₁) r=str.getBytes(v ₁) r=str.getBytes(v ₁)	(>=,<=,>,<) r==i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i] <v r[i]="=v" r[i]<v="" r[i]<v<="" td=""><td>[\s\S]{i₁} [\s\S]{i₁,} [\s\S]{i₁,} [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v-1][\s\S]*</td><td>-1 -1 -1 -1 -1 -1 -1</td><td>NULL NULL NULL NULL NULL NULL NULL NULL</td><td>i₁ i₁ 0 i+1 i+1 i+1 i+1 i+1 i+1</td><td></td></v>	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i ₁ ,} [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v-1][\s\S]*	-1 -1 -1 -1 -1 -1 -1	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1 i+1 i+1 i+1 i+1 i+1	
30	int length() char[] toCharArray()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes(v ₁) r=str.getBytes(v ₁) r=str.getBytes(v ₁)	(>=,<=,>,<)	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i ₁ ,} [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]*	-1 -1 -1 -1 -1 -1 -1 -1 -1	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1 i+1 i+1 i+1 i+1 i+1 i+1	
30	int length() char[] toCharArray() byte[] getBytes(String)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes(v ₁) r=str.getBytes(v ₁) r=str.getBytes(v ₁)	(>=,<=,>,<) r==i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i] <v r[i]="=v" r[i]<v="" r[i]<v<="" td=""><td>[\s\S]{i₁} [\s\S]{i₁,} [\s\S]{i}\v[\s\S]* [\s\S]{i}\v[\s\S]* [\s\S]{i}[\v+1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}\v[\s\S]* [\s\S]{i}[\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}\v[\s\S]* [\s\S]{i}[\v-1][\s\S]* [\s\S]{i}[\v-1][\s\S]* [\s\S]{i}\v[\s\S]* [\s\S]{i}\v[\s\S]*</td><td>-1 -1 -1 -1 -1 -1 -1</td><td>NULL NULL NULL NULL NULL NULL NULL NULL</td><td>i₁ i₁ 0 i+1 i+1 i+1 i+1 i+1 i+1</td><td></td></v>	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i}\v[\s\S]* [\s\S]{i}\v[\s\S]* [\s\S]{i}[\v+1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}\v[\s\S]* [\s\S]{i}[\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}\v[\s\S]* [\s\S]{i}[\v-1][\s\S]* [\s\S]{i}[\v-1][\s\S]* [\s\S]{i}\v[\s\S]* [\s\S]{i}\v[\s\S]*	-1 -1 -1 -1 -1 -1 -1	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1 i+1 i+1 i+1 i+1 i+1	
30	int length() char[] toCharArray() byte[] getBytes(String)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes(v ₁) r=str.getBytes(v ₁) r=str.getBytes(v ₁)	(>=,<=,>,<)	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i ₁ ,} [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]*	-1 -1 -1 -1 -1 -1 -1 -1 -1	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1 i+1 i+1 i+1 i+1 i+1 i+1	
29 30 31 32	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁)	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i}\v[\s\S]* [\s\S]{i}\v[\s\S]* [\s\S]{i}\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}\v-1][\s\S]* [\s\S]{i}\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}\v[\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}\v[\s\S]* [\s\S]{i}\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}\v-1][\s\S]*	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1	
29 30 31 32	int length() char[] toCharArray() byte[] getBytes(String)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁)	(>=,<=,>,<)	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i}v[\s\S]* [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]*	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1	
29 30 31 32 33	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁)	(>=,<=,>,<)	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i}v[\s\S]* [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]*	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1	
29 30 31 32 33	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁)	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i} v[\s\S]* [\s\S]{i} v[\s\S]* [\s\S]{i} [v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i} [v-1][\s\S]* [\s\S]{i} [v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i} [v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i} [v-1][\s\S]* [\s\S]{i} [v-1][\s\S]* [\s\S]{i} [v-1][\s\S]* [\s\S]{i} [v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i} [v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i} [v-1][\s\S]* [\s\S]{i} [v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i} [v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i} [v-1][\s\S]*	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1	
29 30 31 32 33 34 35	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes() String substring(int)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes()	(>=,<=,>,<) r==i ₁ r>=i ₁ r<=i ₁ r(i)==v r(i)>v r(i)==v r(i)>v r(i)==v r(i)=v	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i}v[\s\S]* [\s\S]{i}v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]*	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1	
30 31 32 33 34 35 36	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes() String substring(int) String substring(int,int)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes()	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i]>v r[i]>v r[i]==v r[i]==v r[i]>v r[i]==v r[i]	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i} v[\s\S]* [\s\S]{i} v[\s\S]* [\s\S]{i}[v+1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]* [\s\S]{i}[v-1][\s\S]*	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1	
30 31 32 33 34 35 36 37 38	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes() String substring(int) String substring(int,int) CharSequence subSequence(int,int) String[] split(String) String[] split(String,int)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.substring(i ₁) r=str.substring(i ₁) r=str.subSequence(i ₁ ,i ₂) r=str.split(v) r=str.split(v, i ₁)	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i] <v r[i]="" ="">v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]==v</v>	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i}\v[\s\S]* [\s\S]{i}\v[\s\S]* [\s\S]{i}[\v+1][\s\S]* [\s\S]{i}[\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[\v+1][\s\S]* [\s\S]{i}[\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[\v+1][\s\S]* [\s\S]{i}[\v-1][\s\S]* [\s\S]{i}[\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[\v-1][\s\S]* [\s\S]{i}[\v-1][\s\S]* [\s\S]{i}[\v-1][\s\S]* [\s\S]{i}[\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[\v-1][\s\S]* [\s\S]{i}[\v-1][\s\S]* (convert v+1,v-1 to string first) [\s\S]{i}[\v-1][\s\S]* [\s\S]{i}] [\s\S]{i}] [\s\S]{i}]	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1	
30 31 32 33 34 35 36 37 38 39	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes() String substring(int) String substring(int,int) CharSequence subSequence(int,int) String[] split(String) String[] split(String,int) String replace(char,char)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes()	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]=v	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i}\v[\s\S]*	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1	
30 31 32 33 34 35 36 37 38 39 40	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes() String substring(int) String substring(int,int) CharSequence subSequence(int,int) String[] split(String) String[] split(String,int) String replace(char,char) String replace(CharSequence,CharSequence)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.substring(i ₁) r=str.substring(i ₁ ,i ₂) r=str.subSequence(i ₁ ,i ₂) r=str.split(v) r=str.split(v, i ₁) r=str.replace(v ₁ ,v ₂) r=str.replace(v ₁ ,v ₂)	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i] <v r[i]="" ="">v r[i]==v r[i]>v r[i]==v r[i]>v r[i]<v r[i]="" ="">v r[i]==v r[i]>v r[i]==v r[i]=v r[i]==v r[i]=v r[i]=</v></v>	[\s\S]\{i_1\} [\s\S]\{i_1,\} [\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{i_1\}\[\s\S]* [\s\S]\{i_1\}\[\s\S]* [\s\S]\{i_1\}\[\s\S]* [\s\S]\{i_1\}\[\s\S]* [\s\S]\{i_1\}	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i1 i1 i2 i3 i4 i5 0 0 0 0 0 0 0	
30 31 32 33 34 35 36 37 38 39 40 41	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes() String substring(int) String substring(int,int) CharSequence subSequence(int,int) String[] split(String) String[] split(String,int) String replace(char,char) String replace(CharSequence,CharSequence) String replaceAll(regex,String)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.substring(i ₁) r=str.substring(i ₁) r=str.subSequence(i ₁ ,i ₂) r=str.split(v) r=str.split(v, i ₁) r=str.replace(v ₁ ,v ₂) r=str.replaceAll(v ₁ ,v ₂)	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i]>v r[i]>v r[i]==v r[i]>v r[i]>v r[i] <v r[i]="" ="">v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[</v>	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i}\v[\s\S]*	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i1 i1 i0 i+1 i-1 i-1<	
30 31 32 33 34 35 36 37 38 39 40 41 42	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes() String substring(int) String substring(int,int) CharSequence subSequence(int,int) String[] split(String) String[] split(String,int) String replace(char,char) String replace(CharSequence,CharSequence) String replaceAll(regex,String) String replaceFirst(String, String)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.substring(i ₁) r=str.substring(i ₁ ,i ₂) r=str.subSequence(i ₁ ,i ₂) r=str.split(v) r=str.split(v, i ₁) r=str.replace(v ₁ ,v ₂) r=str.replace(v ₁ ,v ₂)	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i] <v r[i]="" ="">v r[i]==v r[i]>v r[i]==v r[i]>v r[i]<v r[i]="" ="">v r[i]==v r[i]>v r[i]==v r[i]=v r[i]==v r[i]=v r[i]=</v></v>	[\s\S]\{i_1\} [\s\S]\{i_1,\} [\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{i_1\}\[\s\S]* [\s\S]\{i_1\}\[\s\S]* [\s\S]\{i_1\}\[\s\S]* [\s\S]\{i_1\}\[\s\S]* [\s\S]\{i_1\}	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i1 i1 i2 i3 i4 i5 0 0 0 0 0 0 0	
30 31 32 33 34 35 36 37 38 39 40 41 42 43	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes() String substring(int) String substring(int,int) CharSequence subSequence(int,int) String[] split(String) String[] split(String,int) String replace(char,char) String replace(CharSequence,CharSequence) String replaceAll(regex,String)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.substring(i ₁) r=str.substring(i ₁) r=str.substring(i ₁ ,i ₂) r=str.subSequence(i ₁ ,i ₂) r=str.split(v) r=str.split(v, i ₁) r=str.replace(v ₁ ,v ₂) r=str.replaceAll(v ₁ ,v ₂) r=str.replaceFirst(v ₁ ,v ₂) r=str.replaceFirst(v ₁ ,v ₂)	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v	[\s\S]{i ₁ } [\s\S]{i ₁ ,} [\s\S]{i} \v[\s\S]* [\s\S]{i} \v[\s\S]* [\s\S]{i} [\v+1][\s\S]* [\s\S]{i} [\v+1][\s\S]* [\s\S]{i} \v[\s\S]* [\s\S]{i} \v[\s\S]* [\s\S]{i} \v[\s\S]* [\s\S]{i} [\v+1][\s\S]* [\s\S]{i} [\v+1][\s\S]* [\s\S]{i} \v[\s\S]* [\s\S]{i} [\v+1][\s\S]*	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i1 i1 i0 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i-1 i-1<	
30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes() String substring(int) String substring(int,int) CharSequence subSequence(int,int) String[] split(String) String[] split(String,int) String replace(char,char) String replace(CharSequence,CharSequence) String replaceAll(regex,String) String replaceFirst(String, String) String toLowerCase(Locale) String toUpperCase(Locale)	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.substring(i ₁) r=str.substring(i ₁) r=str.substring(i ₁) r=str.split(v) r=str.split(v, i ₁) r=str.replace(v ₁ ,v ₂) r=str.replaceAll(v ₁ ,v ₂) r=str.replaceFirst(v ₁ ,v ₂) r=str.toLowerCase(local) r=str.toUpperCase(local)	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i] <v r[i]="" ="">v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]=v r[i]==v r[i]==v </v>	[\s\S]\{i_1\} [\s\S]\{0,i_1\} [\s\S]\{0,i_1\} [\s\S]\{i\} v[\s\S]* [\s\S]\{i\} v[\s\S]* [\s\S]\{i\} [v-1][\s\S]* [\s\S]\{i\} [v-1][\s\S]* [\s\S]\{i\} [v-1][\s\S]* [\s\S]\{i\} [v-1][\s\S]* [\s\S]\{i\} [v-1][\s\S]* [\s\S]\{i\} v[\s\S]* [\s\S]\{i\} v[\s\S]* [\s\S]\{i\} [v-1][\s\S]* [\s\S]\{i\} [v-1][\s\S]\{i\} [\s\S]\{i\}	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i-1 i-1 i-1 i-1 i-1 i-1 i-1 i-1 i-1 i-	
30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes(Charset) string substring(int) String substring(int,int) CharSequence subSequence(int,int) String[] split(String) String[] split(String,int) String replace(char,char) String replace(CharSequence,CharSequence) String replaceAll(regex,String) String replaceFirst(String, String) String toLowerCase(Locale) String toUpperCase(Locale) String toUpperCase()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.substring(i ₁) r=str.substring(i ₁) r=str.substring(i ₁) r=str.substring(i ₁) r=str.split(v) r=str.split(v) r=str.split(v, i ₁) r=str.replace(v ₁ ,v ₂) r=str.replace(V ₁ ,v ₂) r=str.replaceFirst(v ₁ ,v ₂) r=str.toLowerCase(local) r=str.toUpperCase(local) r=str.toUpperCase()	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]==	[\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{0,i_1\} [\s\S]\{i\}\v[\s\S]* [\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\\\\\\\\\\\\\	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i-1 i-1 i-1 i-1 i-1 i-1 i-1 i-1 i-1 i-	
30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes(Charset) string substring(int) String substring(int,int) CharSequence subSequence(int,int) String[] split(String) String[] split(String,int) String replace(Char,char) String replace(CharSequence,CharSequence) String replaceAll(regex,String) String replaceAll(regex,String) String toLowerCase(Locale) String toUpperCase(Locale) String toUpperCase() String toString()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.substring(i ₁) r=str.substring(i ₁) r=str.substring(i ₁) r=str.substring(i ₁) r=str.split(v) r=str.split(v) r=str.split(v, i ₁) r=str.replace(v ₁ ,v ₂) r=str.replaceAll(v ₁ ,v ₂) r=str.replaceFirst(v ₁ ,v ₂) r=str.toLowerCase() r=str.toUpperCase() r=str.toUpperCase() r=str.toString()	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i] <v r[i]="" ="">v r[i]==v r[i]>v r[i]==v r[i]>v r[i]<v r[i]="" ="">v r[i]==v r[i]>v r[i]==v r[i]=v r[i]</v></v>	[\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{i\} \v[\s\S]* [\s\S]\{i\} \v[\s\S]\{i\} [\s\S]\{i\} \v[\s\S]\{i\} [\s\S]\{i\} \v[\s\S]\{i\} [\s\S]\{i\} \v[\s\S]\{i\} [\s\S]\{i\} \v[\s\S]\{i\} [\s\S]\{i\} [\s\S]\{i\	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i1 i1 i1 i0 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i-1 i-1 </td <td></td>	
30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes(Charset) string substring(int) String substring(int,int) CharSequence subSequence(int,int) String[] split(String) String[] split(String,int) String replace(char,char) String replace(CharSequence,CharSequence) String replaceAll(regex,String) String replaceFirst(String, String) String toLowerCase(Locale) String toUpperCase(Locale) String toUpperCase()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.substring(i ₁) r=str.substring(i ₁) r=str.substring(i ₁) r=str.substring(i ₁) r=str.split(v) r=str.split(v) r=str.split(v, i ₁) r=str.replace(v ₁ ,v ₂) r=str.replace(V ₁ ,v ₂) r=str.replaceFirst(v ₁ ,v ₂) r=str.toLowerCase(local) r=str.toUpperCase(local) r=str.toUpperCase()	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r<=i ₁ r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]>v r[i]==v r[i]==	[\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{0,i_1\} [\s\S]\{i\}\v[\s\S]* [\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\v[\s\S]\{i\}\\\\\\\\\\\\\	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i ₁ i ₁ 0 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i-1 i-1 i-1 i-1 i-1 i-1 i-1 i-1 i-1 i-	
30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 new	int length() char[] toCharArray() byte[] getBytes(String) byte[] getBytes(Charset) byte[] getBytes(Charset) string substring(int) String substring(int,int) CharSequence subSequence(int,int) String[] split(String) String[] split(String,int) String replace(Char,char) String replace(CharSequence,CharSequence) String replaceAll(regex,String) String replaceAll(regex,String) String toLowerCase(Locale) String toUpperCase(Locale) String toUpperCase() String toString()	r=str.length() r=str.length() r=str.length() r=str.toCharArray() r=str.toCharArray() r=str.toCharArray() r=str.getBytes(v ₁) r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.getBytes() r=str.substring(i ₁) r=str.substring(i ₁) r=str.substring(i ₁) r=str.substring(i ₁) r=str.split(v) r=str.split(v, i ₁) r=str.replace(v ₁ ,v ₂) r=str.replaceAll(v ₁ ,v ₂) r=str.replaceFirst(v ₁ ,v ₂) r=str.toLowerCase(local) r=str.toUpperCase(local) r=str.toString() r=str.trim()	(>=,<=,>,<) r==i ₁ r>=i ₁ r>=i ₁ r r==v r[i]==v	[\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{i_1\} [\s\S]\{i\} \v\\\s\S* [\s\S]\{i\} \v\\\\s\S* [\s\S]\{i\} \v\\\\s\S* [\s\S]\{i\} \v\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\	-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -	NULL NULL NULL NULL NULL NULL NULL NULL	i1 i1 i1 i0 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i+1 i-1 i-1 </td <td></td>	