

→ typeof\_TernaryExpression x

```
rule typeof_TernaryExpression {  
  applicable for concept = TernaryExpression as te  
  overrides false  
  
  do {  
    check(typeof(te.condition) ::=: <boolean>);  
    var T;  
    infer typeof(te.thenExpr) :<=: T;  
    infer typeof(te.elseExpr) :<=: T;  
    typeof(te) ::=: T;  
  }  
}
```



typeof\_TernaryExpression

Actions

Refactorings

