

Identificar los lexemas de cada código

```
def area_rectangulo(base,altura):  
    area=base*altura;  
    return area;
```

Lexemas identificados:

1. Palabra clave: def
2. Identificador de función: area\_rectangulo
3. Paréntesis: ( y )
4. Identificadores de parámetros: base, altura
5. Dos puntos: :
6. Identificador de variable: area
7. Operador de asignación: =
8. Identificadores: base, altura
9. Operador de multiplicación: \*
10. Punto y coma: ;
11. Palabra clave: return
12. Identificador: area
13. Punto y coma: ;

```
fun f_1(x) = let val x=7  
in  
    (let val x=3 in x-1 end) + (let val y=x+4 in y+2 end)  
end
```

Lexemas identificados:

1. Palabra clave: fun

2. Identificador de función: f\_1
3. Paréntesis: ( y )
4. Identificador de parámetro: x
5. Operador de asignación: =
6. Palabra clave: let
7. Palabra clave: val
8. Identificador: x
9. Operador de asignación: =
10. Literal numérico: 7
11. Palabra clave: in
12. Paréntesis: ( y )
13. Palabra clave: let
14. Palabra clave: val
15. Identificador: x
16. Operador de asignación: =
17. Literal numérico: 3
18. Palabra clave: in
19. Identificador: x
20. Operador de resta: -
21. Literal numérico: 1
22. Palabra clave: end
23. Operador de suma: +
24. Paréntesis: ( y )
25. Palabra clave: let
26. Palabra clave: val
27. Identificador: y
28. Operador de asignación: =
29. Identificador: x
30. Operador de suma: +
31. Literal numérico: 4
32. Palabra clave: in
33. Identificador: y
34. Operador de suma: +

35. Literal numérico: 2

36. Palabra clave: end

37. Palabra clave: end

```
int x = 10**3//4/-557.45-668.56>80?:1;
```

Lexemas identificados:

1. Tipo de dato: int
2. Identificador: x
3. Operador de asignación: =
4. Literal numérico: 10
5. Operador de exponenciación (o multiplicación doble): \*\*
6. Literal numérico: 3
7. Operador de comentario de línea (o división doble): //
8. Literal numérico: 4
9. Operador de división: /
10. Operador de negación (signo menos): -
11. Literal numérico (punto flotante): 557.45
12. Operador de resta: -
13. Literal numérico (punto flotante): 668.56
14. Operador de comparación mayor que: >
15. Literal numérico: 80
16. Operador ternario (signo de interrogación): ?
17. Operador ternario (dos puntos): :
18. Literal numérico: 1
19. Punto y coma: ;