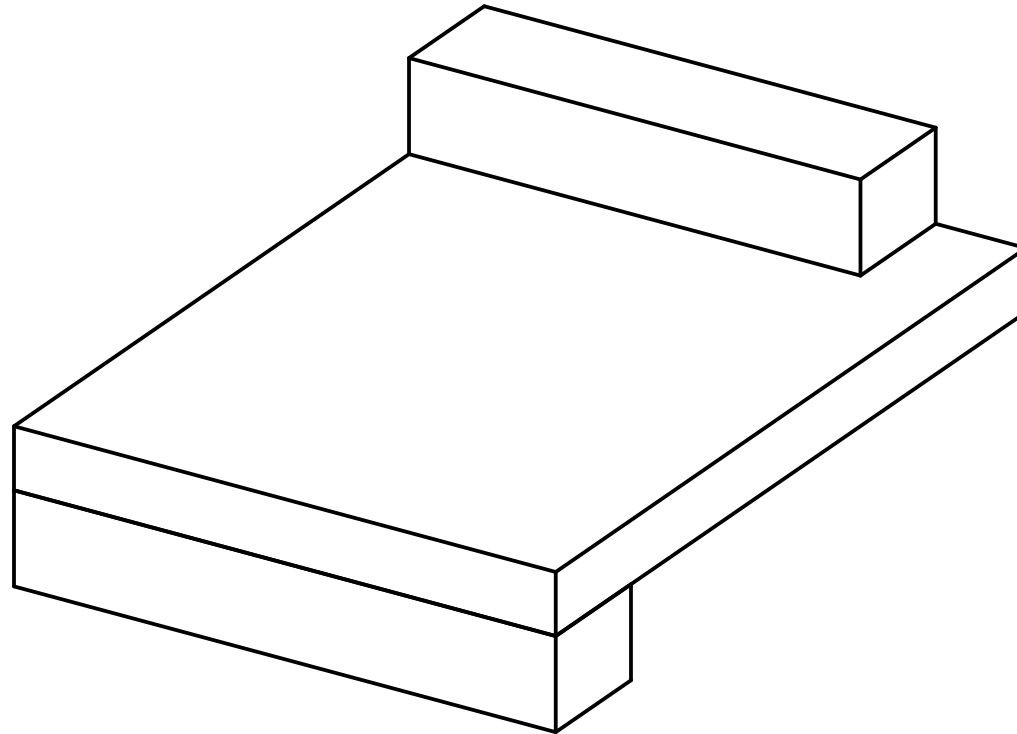


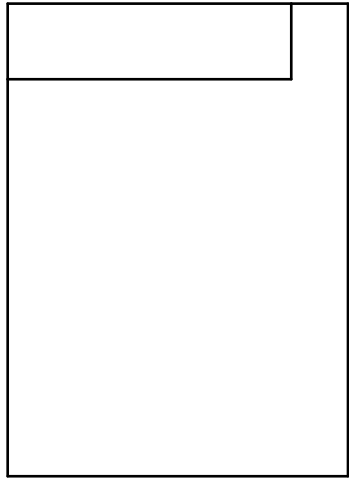
Bench hook



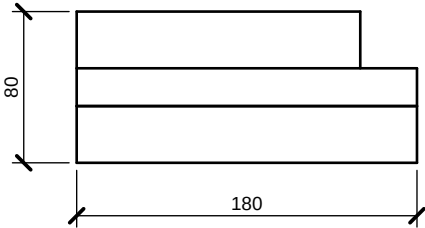
parts	3
overall	180 x 250 x 80 (W x D x H)
material	oak
date	2026-10-25
made with	woodwright 0.9.0

Contents

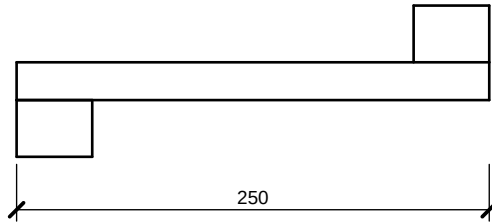
Drawings	3
Assembly	3
Parts	4
1 base	5
2 fence	6
3 hook	7
 Cut list	 8
 Appendix: source	 9



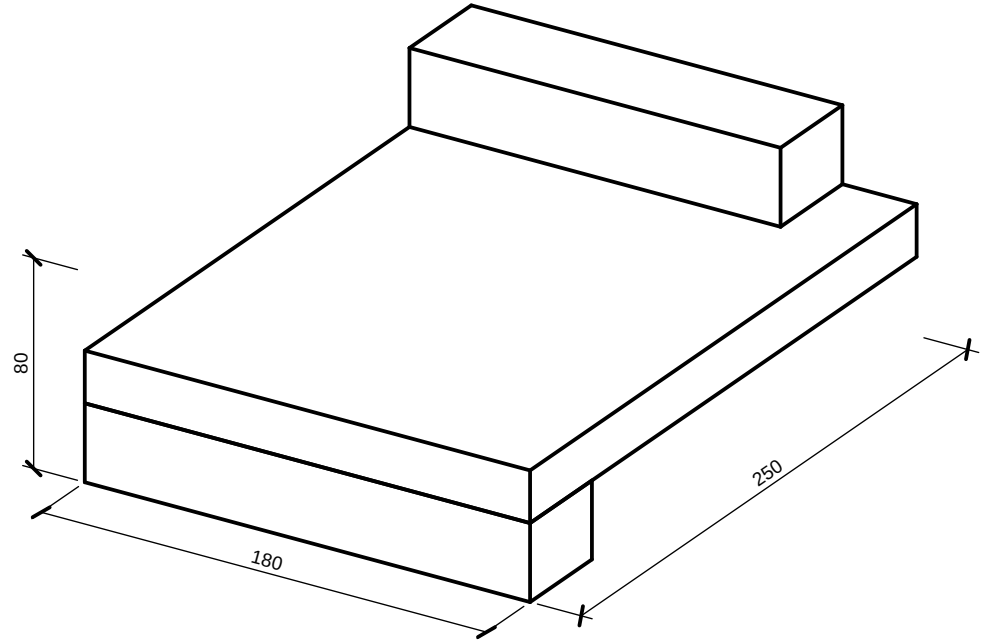
TOP



FRONT

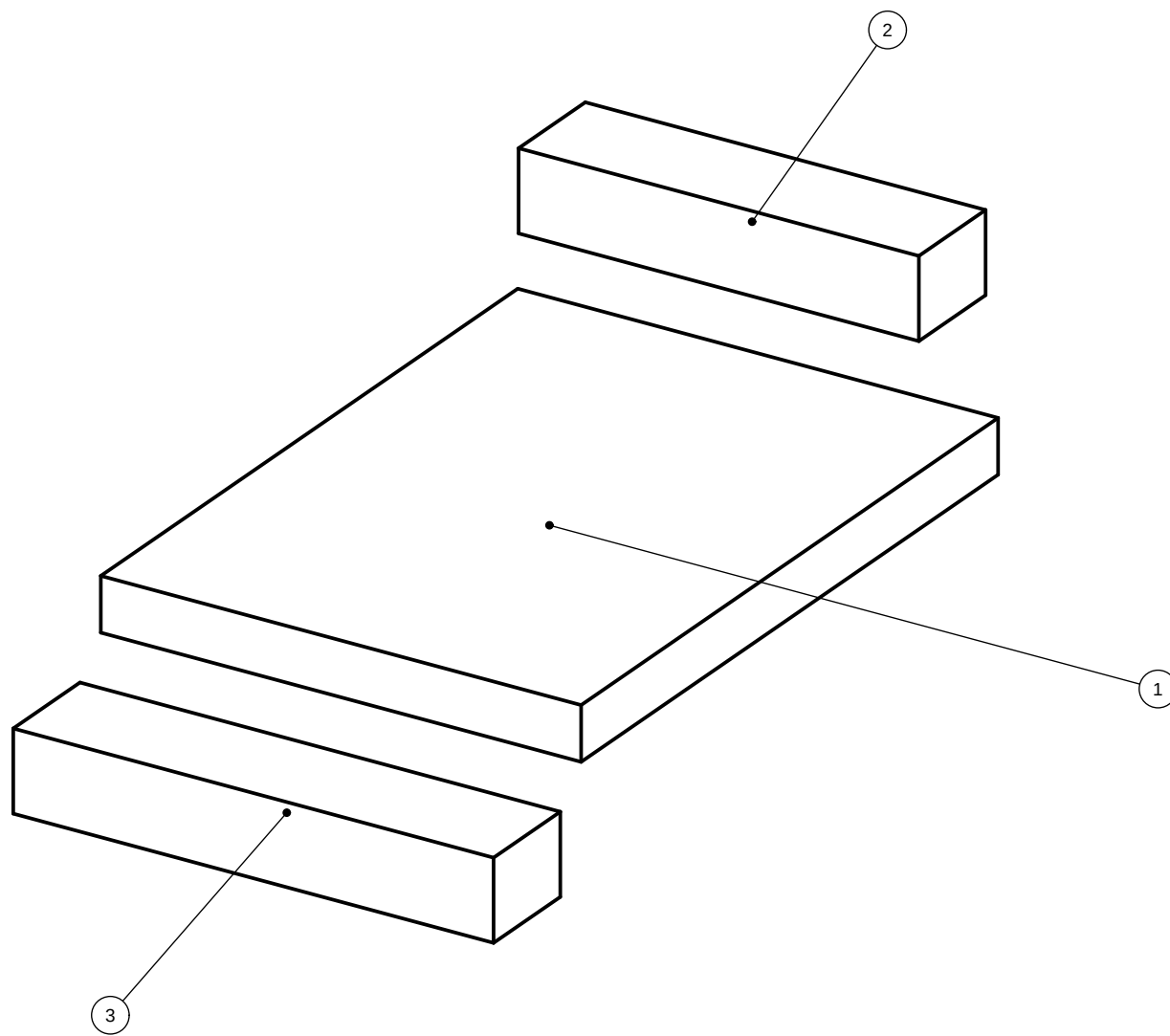


RIGHT SIDE



ISOMETRIC

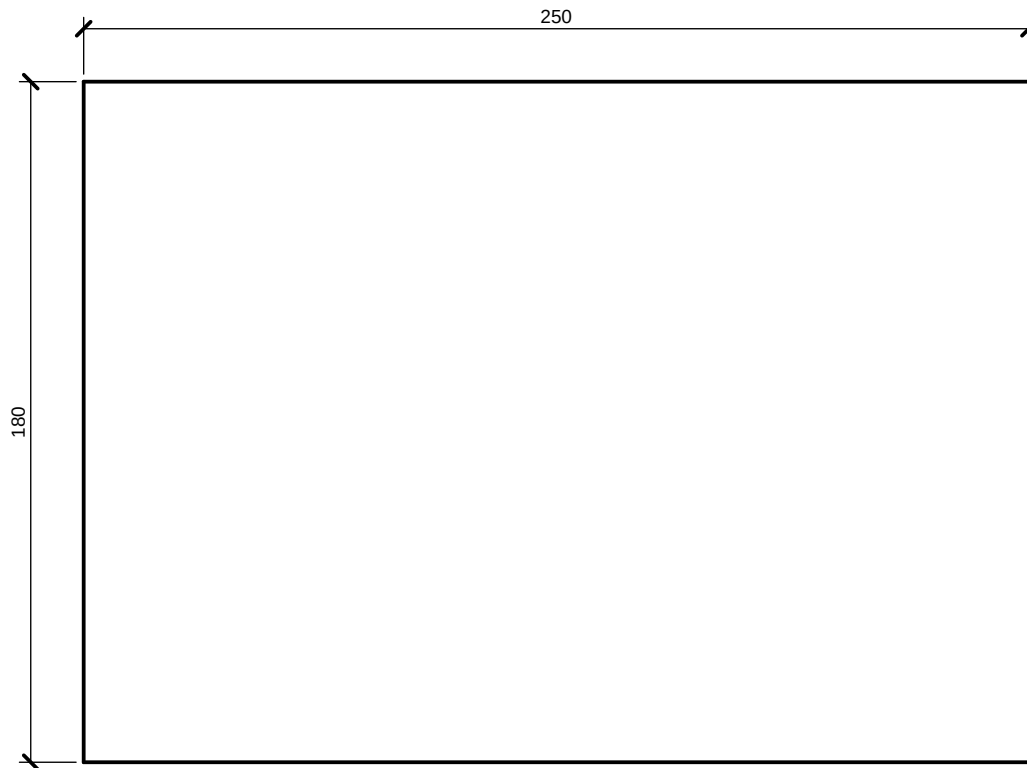
assembly	bench_hook
parts	3
overall	180 x 250 x 80 (W x D x H)
material	oak
scale	1:4, mm



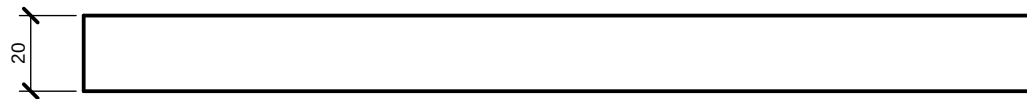
EXPLODED

no.	part	qty	cut size (L x W x T)	material
1	base	1	250 x 180 x 20	oak
2	fence	1	150 x 40 x 30	oak
3	hook	1	180 x 40 x 30	oak

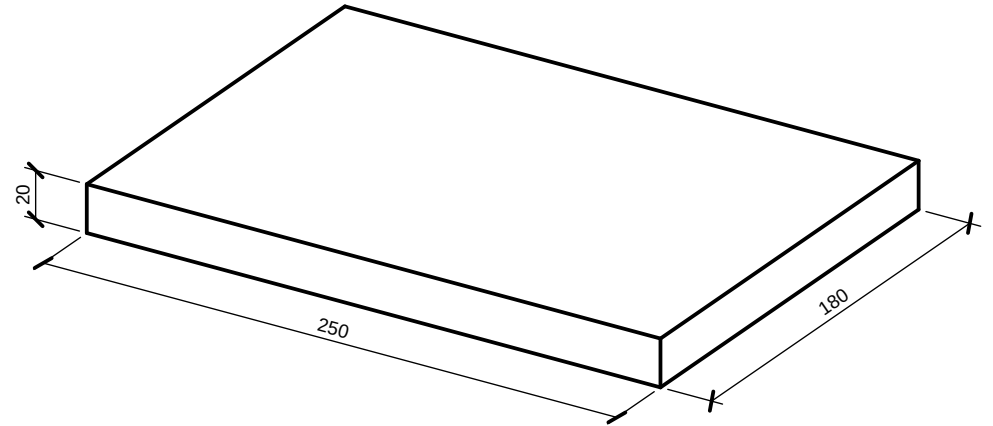
assembly	bench_hook
parts	3
overall	180 x 250 x 80 (W x D x H)
material	oak
scale	not to scale, mm



TOP FACE

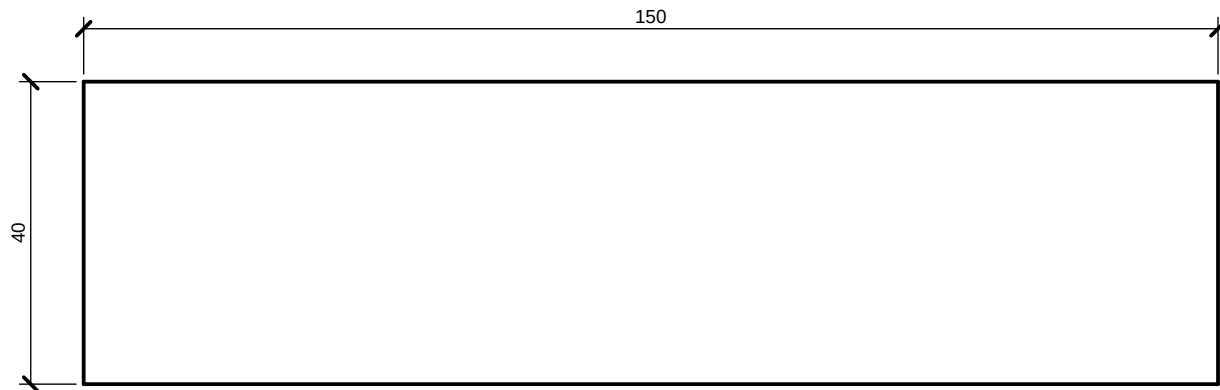


FRONT EDGE

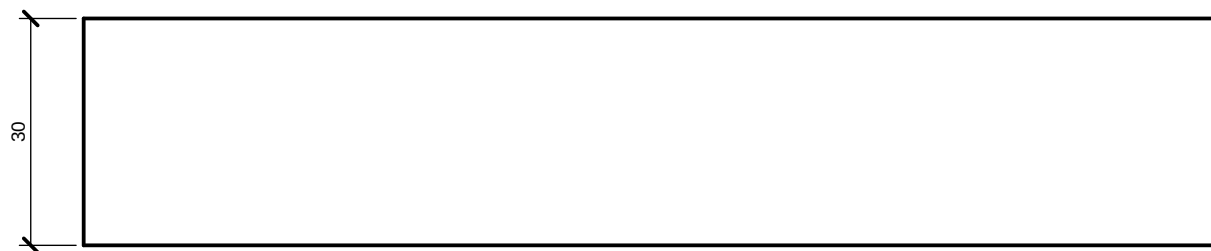


ISOMETRIC

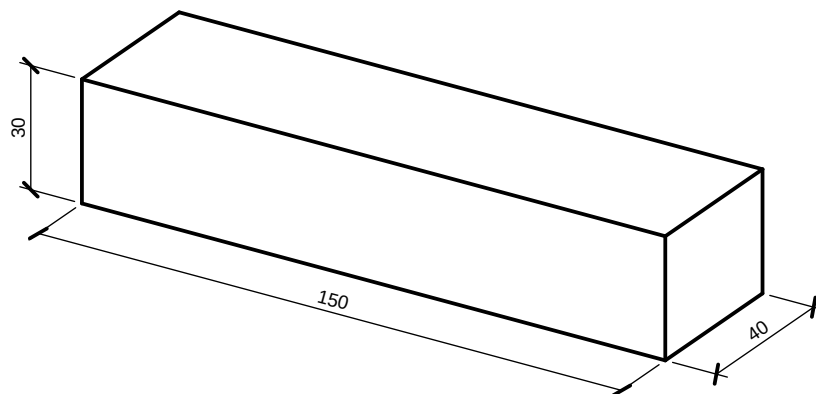
part	base
of	bench_hook
make	1
cut size	250 x 180 x 20
material	oak
scale	1:2, mm



TOP FACE

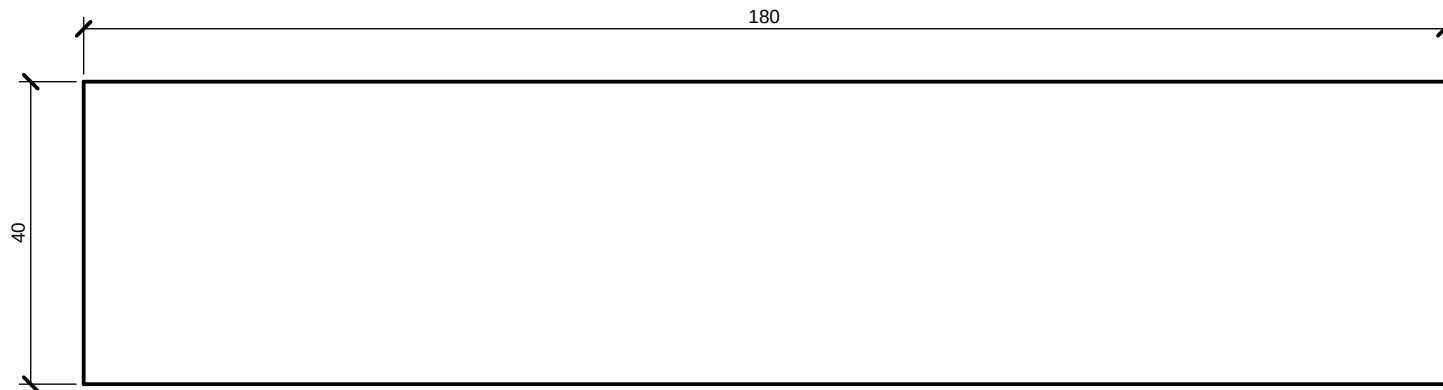


FRONT EDGE

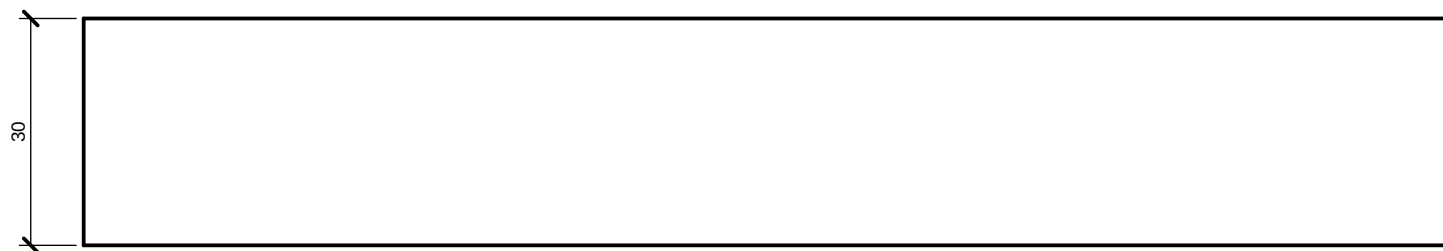


ISOMETRIC

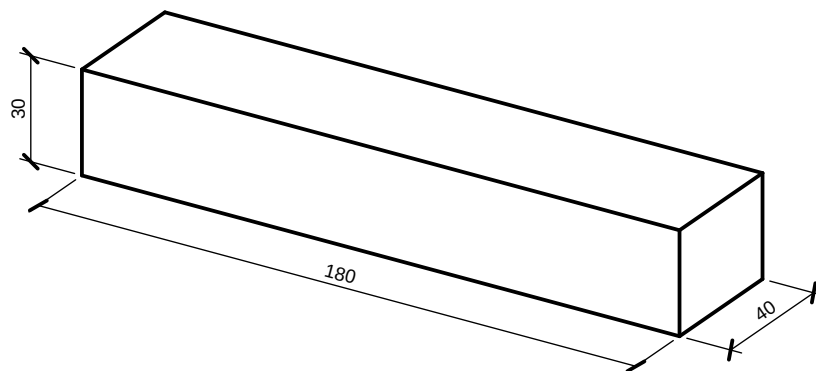
part	fence
of	bench_hook
make	1
cut size	150 x 40 x 30
material	oak
scale	1:1, mm



TOP FACE



FRONT EDGE



ISOMETRIC

part	hook
of	bench_hook
make	1
cut size	180 x 40 x 30
material	oak
scale	1:1, mm

Cut list

base	250 x 180 x 20 mm	oak
fence	150 x 40 x 30 mm	oak
hook	180 x 40 x 30 mm	oak

Appendix: source

Made with woodwright 0.9.0, build123d 0.13.0, Python 3.14.4.

To make these plans again:

```
uvx --python 3.14 woodwright==0.9.0 export projects.bench_hook
```

projects.bench_hook, in woodwright/examples/projects.py

```
def bench_hook(
    length: float = 250,
    width: float = 180,
    fence_length: float = 150,
    fence_width: float = 40,
    stock: Stock = OAK,
    fence_stock: Stock = THICK_OAK,
) -> Assembly:
    """A bench hook: a base with a fence on top at the back and a hook below
    at the front, glued face to face across the grain.

    The base's grain runs away from the woodworker. The fence stops short of
    the right side, so the saw cuts into the base, not the fence; the hook
    spans the whole width.
    """
    # The base lies flat, its length away from the viewer: its back edge is
    # to the left.
    asm = Assembly("bench_hook", root_location=orient(right_end=BACK, ref_face=UP))
    base = asm.add(Board("base", stock, length=length, width=width))
    fence = asm.add(Board("fence", fence_stock, length=fence_length, width=fence_width))
    hook = asm.add(Board("hook", fence_stock, length=width, width=fence_width))
    asm.join(
        Butt(
            base.top_face,
            fence.bottom_face,
            flush=(base.right_end, fence.back_edge),
            position=flush_with(base.back_edge),
        )
    )
    asm.join(Butt(base.bottom_face, hook.top_face, flush=(base.left_end, hook.front_edge)))
```

```
return asm
```