

Pict Snip: Build Snips from Picts

Version 9.0.0.4

November 20, 2025

```
(require pict/snip)      package: pict-snip-lib
```

The `pict/snip` library constructs `snip%` instances that draw based on a given `pict?` object.

```
pict-snip% : class?  
  superclass: snip%
```

```
(new pict-snip% [pict pict]) → (is-a?/c pict-snip%)  
  pict : pict?
```

Creates a `pict-snip%` object, using `pict` to draw.

```
(send a-pict-snip get-pict) → pict?
```

Returns the pict passed to the constructor.

```
(send a-pict-snip get-extent dc  
  x  
  y  
  w  
  h  
  descent  
  lspace  
  rspace) → void?  
  
dc : (is-a?/c dc<%>)  
x : real?  
y : real?  
w : (or/c (box/c (and/c real? (not/c negative?))) #f)  
h : (or/c (box/c (and/c real? (not/c negative?))) #f)
```

```

descent : (or/c (box/c (and/c real? (not/c negative?))) #f)
lspace : (or/c (box/c (and/c real? (not/c negative?))) #f)
rspace : (or/c (box/c (and/c real? (not/c negative?))) #f)

```

Updates the arguments based on the size of the pict returned from `get-pict`.

```

(send a-pict-snip draw dc
      x
      y
      left
      top
      right
      bottom
      dx
      dy
      draw-caret) → void?

dc : (is-a?/c dc<%>)
x : real?
y : real?
left : real?
top : real?
right : real?
bottom : real?
dx : real?
dy : real?
draw-caret : (or/c 'no-caret 'show-inactive-caret 'show-caret
                  (cons/c exact-nonnegative-integer?
                          exact-nonnegative-integer?))

```

Overrides `draw` in `snip%`.

Draws the pict returned from `get-pict`.

```

(send a-pict-snip write f) → void?
f : (is-a?/c editor-stream-out%)

```

Uses a `record-dc%` to write the way that the result of `get-pict` draws and saves that to `f`.

```

(send a-pict-snip copy) → (is-a?/c pict-snip%)

```

Returns a `snip` that has the same pict as this one.

```

snipclass : (is-a?/c snip-class%)

```

The `snip-class%` instance used by instances of `pict-snip%`.

```
| reader : (is-a?/c snip-reader<%>)
```

The `snip-reader<%>` instance used by the `wxme` library.