polyn-immutable

```
{
  "suiteId": "SDA2A75",
  "time": 71804.9998395145,
  "type": "START",
  "plan": {
    "count": 40,
    "completed": 0,
    "order": [
      "T2794269929",
      "T115500802",
      "T1412403848",
      "T3774403453",
      "T3839474028",
      "T3570737992",
      "T3248676982",
      "T3720816098",
      "T3378731568",
      "T1809451320",
      "T1868228119",
      "T3558144078",
      "T3978851634",
      "T2117899464",
      "T4162694622",
      "T2783448395",
      "T1134632789",
      "T2289548824",
      "T2681499153",
      "T3598350111",
      "T830319501",
      "T2354375029",
      "T2789853614",
      "T2676344648",
      "T775081587",
      "T930823784",
      "T3908645753",
      "T2542537529",
      "T176323384",
      "T4148801186",
      "T331286067",
      "T1131854492",
      "T647668896",
      "T2355015786",
      "T2937703443",
      "T3044950036",
      "T3937129236",
      "T34662498",
      "T1772791911",
      "T2036520937"
    ]
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2794269929",
  "time": 128004.99983131886,
  "type": "START_TEST",
  "behavior": "given `immutable`, when initialized with a valid name, it should return an instanceof ValidatedImmutable",
  "behaviors": [
    "given `immutable`",
    "when initialized with a valid name",
    "it should return an instanceof ValidatedImmutable"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T115500802",
  "time": 128929.99988980591,
  "type": "START_TEST",
  "behavior": "given `immutable`, when initialized with a valid name, it should return a constructor",
  "behaviors": [
    "given `immutable`",
    "when initialized with a valid name",
    "it should return a constructor"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2794269929",
  "count": 1,
  "time": 136189.99999016523,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when initialized with a valid name, it should return an instanceof ValidatedImmutable",
  "behaviors": [
    "given `immutable`",
    "when initialized with a valid name",
```

```json
      "it should return an instanceof ValidatedImmutable"
    ],
    "duration": {
      "given": {
        "seconds": 0.000014999881386756897,
        "milliseconds": 0.014999881386756897,
        "microseconds": 14.999881386756897,
        "nanoseconds": 14999.881386756897
      },
      "when": {
        "seconds": 0.0004999998491257429,
        "milliseconds": 0.4999998491257429,
        "microseconds": 499.9998491257429,
        "nanoseconds": 499999.8491257429
      },
      "then": {
        "seconds": 0.006720000179484487,
        "milliseconds": 6.720000179484487,
        "microseconds": 6720.000179484487,
        "nanoseconds": 6720000.179484487
      },
      "total": {
        "seconds": 0.007234999909996986,
        "milliseconds": 7.234999909996986,
        "microseconds": 7234.999909996986,
        "nanoseconds": 7234999.909996986
      }
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T115500802",
  "count": 2,
  "time": 137334.99986119568,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when initialized with a valid name, it should return a constructor",
  "behaviors": [
    "given `immutable`",
    "when initialized with a valid name",
    "it should return a constructor"
  ],
  "duration": {
    "given": {
      "seconds": 0.000014999881386756897,
      "milliseconds": 0.014999881386756897,
      "microseconds": 14.999881386756897,
      "nanoseconds": 14999.881386756897
    },
    "when": {
      "seconds": 0.0004999998491257429,
      "milliseconds": 0.4999998491257429,
      "microseconds": 499.9998491257429,
      "nanoseconds": 499999.8491257429
    },
    "then": {
      "seconds": 0.007865000050514936,
      "milliseconds": 7.8650000505149364,
      "microseconds": 7865.000050514936,
      "nanoseconds": 7865000.050514936
    },
    "total": {
      "seconds": 0.008379999781027436,
      "milliseconds": 8.379999781027436,
      "microseconds": 8379.999781027436,
      "nanoseconds": 8379999.781027436
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T1412403848",
  "time": 138869.99990791082,
  "type": "START_TEST",
  "behavior": "given `immutable`, when initialized with an invalid name, it should throw",
  "behaviors": [
    "given `immutable`",
    "when initialized with an invalid name",
    "it should throw"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T1412403848",
  "count": 3,
  "time": 139770.0000088662,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when initialized with an invalid name, it should throw",
  "behaviors": [
```

```json
      "given `immutable`",
      "when initialized with an invalid name",
      "it should throw"
    ],
    "duration": {
      "given": {
        "seconds": 0,
        "milliseconds": 0,
        "microseconds": 0,
        "nanoseconds": 0
      },
      "when": {
        "seconds": -1,
        "milliseconds": -1,
        "microseconds": -1,
        "nanoseconds": -1
      },
      "then": {
        "seconds": 0.00013500009663403034,
        "milliseconds": 0.13500009663403034,
        "microseconds": 135.00009663403034,
        "nanoseconds": 135000.09663403034
      },
      "total": {
        "seconds": 0.00013500009663403034,
        "milliseconds": 0.13500009663403034,
        "microseconds": 135.00009663403034,
        "nanoseconds": 135000.09663403034
      }
    }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3774403453",
  "time": 141029.9998242408,
  "type": "START_TEST",
  "behavior": "given `immutable`, when initialized with an existing blueprint, it should return the value",
  "behaviors": [
    "given `immutable`",
    "when initialized with an existing blueprint",
    "it should return the value"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3774403453",
  "count": 4,
  "time": 143135.00002026558,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when initialized with an existing blueprint, it should return the value",
  "behaviors": [
    "given `immutable`",
    "when initialized with an existing blueprint",
    "it should return the value"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": 0.0001750001683831215,
      "milliseconds": 0.1750001683831215,
      "microseconds": 175.0001683831215,
      "nanoseconds": 175000.1683831215
    },
    "then": {
      "seconds": 0.001275000162422657,
      "milliseconds": 1.275000162422657,
      "microseconds": 1275.000162422657,
      "nanoseconds": 1275000.162422657
    },
    "total": {
      "seconds": 0.001450000330857785,
      "milliseconds": 1.450000330857785,
      "microseconds": 1450.0003308057785,
      "nanoseconds": 1450000.3308057785
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3839474028",
  "time": 144299.99981075525,
  "type": "START_TEST",
  "behavior": "given `immutable`, when initialized with an invalid blueprint, it should throw",
  "behaviors": [
```

```json
      "given `immutable`",
      "when initialized with an invalid blueprint",
      "it should throw"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3839474028",
  "count": 5,
  "time": 145929.99988235533,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when initialized with an invalid blueprint, it should throw",
  "behaviors": [
    "given `immutable`",
    "when initialized with an invalid blueprint",
    "it should throw"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": -1,
      "milliseconds": -1,
      "microseconds": -1,
      "nanoseconds": -1
    },
    "then": {
      "seconds": 0.0007599999662488699,
      "milliseconds": 0.7599999662488699,
      "microseconds": 759.9999662488699,
      "nanoseconds": 759999.9662488699
    },
    "total": {
      "seconds": 0.0007599999662488699,
      "milliseconds": 0.7599999662488699,
      "microseconds": 759.9999662488699,
      "nanoseconds": 759999.9662488699
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3570737992",
  "time": 147149.9998588115,
  "type": "START_TEST",
  "behavior": "given `immutable`, when initialized without a name, it should throw",
  "behaviors": [
    "given `immutable`",
    "when initialized without a name",
    "it should throw"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3570737992",
  "count": 6,
  "time": 148274.99981038272,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when initialized without a name, it should throw",
  "behaviors": [
    "given `immutable`",
    "when initialized without a name",
    "it should throw"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": -1,
      "milliseconds": -1,
      "microseconds": -1,
      "nanoseconds": -1
    },
    "then": {
      "seconds": 0.00019499985501170158,
      "milliseconds": 0.19499985501170158,
      "microseconds": 194.99985501170158,
      "nanoseconds": 194999.85501170158
    },
    "total": {
```

```
        "seconds": 0.00019499985501170158,
        "milliseconds": 0.19499985501170158,
        "microseconds": 194.99985501170158,
        "nanoseconds": 194999.85501170158
      }
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3248676982",
  "time": 149770.00001817942,
  "type": "START_TEST",
  "behavior": "given `immutable`, when initialized with a validator that intercepts values, it should use the intercepted value
  "behaviors": [
    "given `immutable`",
    "when initialized with a validator that intercepts values",
    "it should use the intercepted values"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3248676982",
  "count": 7,
  "time": 150930.00000342727,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when initialized with a validator that intercepts values, it should use the intercepted valu
  "behaviors": [
    "given `immutable`",
    "when initialized with a validator that intercepts values",
    "it should use the intercepted values"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": 0.000260000117123127,
      "milliseconds": 0.260000117123127,
      "microseconds": 260.000117123127,
      "nanoseconds": 260000.11712312698
    },
    "then": {
      "seconds": 0.00011000013910233974,
      "milliseconds": 0.11000013910233974,
      "microseconds": 110.00013910233974,
      "nanoseconds": 110000.13910233974
    },
    "total": {
      "seconds": 0.00037000025622546673,
      "milliseconds": 0.37000025622546673,
      "microseconds": 370.0002562254667,
      "nanoseconds": 370000.2562254667
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3720816098",
  "time": 152534.999884665,
  "type": "START_TEST",
  "behavior": "given `immutable`, when initialized with functionsOnPrototype option, it should be comparable with deep equals"
  "behaviors": [
    "given `immutable`",
    "when initialized with functionsOnPrototype option",
    "it should be comparable with deep equals"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3720816098",
  "count": 8,
  "time": 154020.00001631677,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when initialized with functionsOnPrototype option, it should be comparable with deep equals"
  "behaviors": [
    "given `immutable`",
    "when initialized with functionsOnPrototype option",
    "it should be comparable with deep equals"
  ],
  "duration": {
    "given": {
      "seconds": 0.0000500003807246685,
      "milliseconds": 0.00500003807246685,
      "microseconds": 5.00003807246685,
```

```
      "nanoseconds": 5000.03807246685
    },
    "when": {
      "seconds": 0.0003099997993558645,
      "milliseconds": 0.3099997993558645,
      "microseconds": 309.9997993558645,
      "nanoseconds": 309999.7993558645
    },
    "then": {
      "seconds": 0.00036000018008053303,
      "milliseconds": 0.360000180080533,
      "microseconds": 360.000180080533,
      "nanoseconds": 360000.180080533
    },
    "total": {
      "seconds": 0.0006750000175088644,
      "milliseconds": 0.6750000175088644,
      "microseconds": 675.0000175088644,
      "nanoseconds": 675000.0175088644
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3378731568",
  "time": 157334.99987982213,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an immutable is constructed with valid input, it should return the value",
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with valid input",
    "it should return the value"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T1809451320",
  "time": 158734.99982990324,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an immutable is constructed with valid input, it should return a class that supports `i
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with valid input",
    "it should return a class that supports `instanceof`"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T1868228119",
  "time": 160304.999910295,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the primitives, recursiv
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with valid input",
    "it should freeze the primitives, recursively (strict mode)"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3558144078",
  "time": 161764.99985158443,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the primitives, recursiv
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with valid input",
    "it should freeze the primitives, recursively"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3978851634",
  "time": 163234.9998690188,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the functions",
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with valid input",
    "it should freeze the functions"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2117899464",
  "time": 164484.99984107912,
  "type": "START_TEST",
```

    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should clone functions defined in anot
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should clone functions defined in another scope"
    ]
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T4162694622",
    "time": 165684.99989807606,
    "type": "START_TEST",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should clone functions defined in anot
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should clone functions defined in another scope (strict mode)"
    ]
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T2783448395",
    "time": 166904.99987453222,
    "type": "START_TEST",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the objects",
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should freeze the objects"
    ]
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T1134632789",
    "time": 168214.99983780086,
    "type": "START_TEST",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the arrays",
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should freeze the arrays"
    ]
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T2289548824",
    "time": 169524.9998010695,
    "type": "START_TEST",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the values inside the ar
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should freeze the values inside the arrays"
    ]
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T2681499153",
    "time": 170794.9999254197,
    "type": "START_TEST",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should NOT allow to mutate array conte
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should NOT allow to mutate array contents"
    ]
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T3598350111",
    "time": 172509.99994575977,
    "type": "START_TEST",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should not allow property deletion",
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should not allow property deletion"
    ]
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T830319501",
    "time": 174054.9998357892,
    "type": "START_TEST",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should not allow property deletion (st
    "behaviors": [

```json
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should not allow property deletion (strict mode)"
    ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3378731568",
  "count": 9,
  "time": 181414.99999910593,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an immutable is constructed with valid input, it should return the value",
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with valid input",
    "it should return the value"
  ],
  "duration": {
    "given": {
      "seconds": 0.00000500003807246685,
      "milliseconds": 0.00500003807246685,
      "microseconds": 5.00003807246685,
      "nanoseconds": 5000.03807246685
    },
    "when": {
      "seconds": 0.0018549999222159386,
      "milliseconds": 1.8549999222159386,
      "microseconds": 1854.9999222159386,
      "nanoseconds": 1854999.9222159386
    },
    "then": {
      "seconds": 0.005684999981895089,
      "milliseconds": 5.684999981895089,
      "microseconds": 5684.999981895089,
      "nanoseconds": 5684999.981895089
    },
    "total": {
      "seconds": 0.007544999942183495,
      "milliseconds": 7.544999942183495,
      "microseconds": 7544.999942183495,
      "nanoseconds": 7544999.942183495
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T1809451320",
  "count": 10,
  "time": 183184.99997258186,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an immutable is constructed with valid input, it should return a class that supports `i:
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with valid input",
    "it should return a class that supports `instanceof`"
  ],
  "duration": {
    "given": {
      "seconds": 0.00000500003807246685,
      "milliseconds": 0.00500003807246685,
      "microseconds": 5.00003807246685,
      "nanoseconds": 5000.03807246685
    },
    "when": {
      "seconds": 0.0018549999222159386,
      "milliseconds": 1.8549999222159386,
      "microseconds": 1854.9999222159386,
      "nanoseconds": 1854999.9222159386
    },
    "then": {
      "seconds": 0.007454999955371022,
      "milliseconds": 7.454999955371022,
      "microseconds": 7454.999955371022,
      "nanoseconds": 7454999.955371022
    },
    "total": {
      "seconds": 0.009314999915659428,
      "milliseconds": 9.314999915659428,
      "microseconds": 9314.999915659428,
      "nanoseconds": 9314999.915659428
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T1868228119",
  "count": 11,
  "time": 184929.9999885261,
```

```json
    "type": "TEST",
    "status": "PASSED",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the primitives, recursiv
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should freeze the primitives, recursively (strict mode)"
    ],
    "duration": {
      "given": {
        "seconds": 0.00000500003807246685,
        "milliseconds": 0.00500003807246685,
        "microseconds": 5.00003807246685,
        "nanoseconds": 5000.03807246685
      },
      "when": {
        "seconds": 0.0018549999222159386,
        "milliseconds": 1.8549999222159386,
        "microseconds": 1854.9999222159386,
        "nanoseconds": 1854999.9222159386
      },
      "then": {
        "seconds": 0.009199999971315265,
        "milliseconds": 9.199999971315265,
        "microseconds": 9199.999971315265,
        "nanoseconds": 9199999.971315265
      },
      "total": {
        "seconds": 0.01105999993160367,
        "milliseconds": 11.05999993160367,
        "microseconds": 11059.99993160367,
        "nanoseconds": 11059999.93160367
      }
    }
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T2117899464",
    "count": 12,
    "time": 186639.99997079372,
    "type": "TEST",
    "status": "PASSED",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should clone functions defined in anot
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should clone functions defined in another scope"
    ],
    "duration": {
      "given": {
        "seconds": 0.00000500003807246685,
        "milliseconds": 0.00500003807246685,
        "microseconds": 5.00003807246685,
        "nanoseconds": 5000.03807246685
      },
      "when": {
        "seconds": 0.0018549999222159386,
        "milliseconds": 1.8549999222159386,
        "microseconds": 1854.9999222159386,
        "nanoseconds": 1854999.9222159386
      },
      "then": {
        "seconds": 0.01090500014834106,
        "milliseconds": 10.90500014834106,
        "microseconds": 10905.00014834106,
        "nanoseconds": 10905000.14834106
      },
      "total": {
        "seconds": 0.012765000108629465,
        "milliseconds": 12.765000108629465,
        "microseconds": 12765.000108629465,
        "nanoseconds": 12765000.108629465
      }
    }
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T4162694622",
    "count": 13,
    "time": 188414.99998234212,
    "type": "TEST",
    "status": "PASSED",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should clone functions defined in anot
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should clone functions defined in another scope (strict mode)"
    ],
    "duration": {
      "given": {
        "seconds": 0.00000500003807246685,
```

```json
      "milliseconds": 0.00500003807246685,
      "microseconds": 5.00003807246685,
      "nanoseconds": 5000.03807246685
    },
    "when": {
      "seconds": 0.0018549999222159386,
      "milliseconds": 1.8549999222159386,
      "microseconds": 1854.9999222159386,
      "nanoseconds": 1854999.9222159386
    },
    "then": {
      "seconds": 0.01268000015988946,
      "milliseconds": 12.68000015988946,
      "microseconds": 12680.00015988946,
      "nanoseconds": 12680000.15988946
    },
    "total": {
      "seconds": 0.014540000120177865,
      "milliseconds": 14.540000120177865,
      "microseconds": 14540.000120177865,
      "nanoseconds": 14540000.120177865
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2681499153",
  "count": 14,
  "time": 190274.99994263053,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an immutable is constructed with valid input, it should NOT allow to mutate array conten
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with valid input",
    "it should NOT allow to mutate array contents"
  ],
  "duration": {
    "given": {
      "seconds": 0.00000500003807246685,
      "milliseconds": 0.00500003807246685,
      "microseconds": 5.00003807246685,
      "nanoseconds": 5000.03807246685
    },
    "when": {
      "seconds": 0.0018549999222159386,
      "milliseconds": 1.8549999222159386,
      "microseconds": 1854.9999222159386,
      "nanoseconds": 1854999.9222159386
    },
    "then": {
      "seconds": 0.014535000082105398,
      "milliseconds": 14.535000082105398,
      "microseconds": 14535.000082105398,
      "nanoseconds": 14535000.082105398
    },
    "total": {
      "seconds": 0.016395000042393804,
      "milliseconds": 16.395000042393804,
      "microseconds": 16395.000042393804,
      "nanoseconds": 16395000.042393804
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T830319501",
  "count": 15,
  "time": 192249.9998472631,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an immutable is constructed with valid input, it should not allow property deletion (st
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with valid input",
    "it should not allow property deletion (strict mode)"
  ],
  "duration": {
    "given": {
      "seconds": 0.00000500003807246685,
      "milliseconds": 0.00500003807246685,
      "microseconds": 5.00003807246685,
      "nanoseconds": 5000.03807246685
    },
    "when": {
      "seconds": 0.0018549999222159386,
      "milliseconds": 1.8549999222159386,
      "microseconds": 1854.9999222159386,
      "nanoseconds": 1854999.9222159386
    },
    "then": {
```

          "seconds": 0.016509999986737967,
          "milliseconds": 16.509999986737967,
          "microseconds": 16509.999986737967,
          "nanoseconds": 16509999.986737967
        },
        "total": {
          "seconds": 0.018369999947026372,
          "milliseconds": 18.369999947026372,
          "microseconds": 18369.999947026372,
          "nanoseconds": 18369999.947026372
        }
      }
    }
  {
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T3558144078",
    "count": 16,
    "time": 194679.99995686114,
    "type": "TEST",
    "status": "FAILED",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the primitives, recursive
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should freeze the primitives, recursively"
    ],
    "error": {
      "message": "Cannot assign to read only property 'requiredString' of object '#<ValidatedImmutable>'",
      "stack": "TypeError: Cannot assign to read only property 'requiredString' of object '#<ValidatedImmutable>'\n   at http:/
    }
  }
  {
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T3978851634",
    "count": 17,
    "time": 197694.99986432493,
    "type": "TEST",
    "status": "FAILED",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the functions",
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should freeze the functions"
    ],
    "error": {
      "message": "Cannot assign to read only property 'func' of object '#<ValidatedImmutable>'",
      "stack": "TypeError: Cannot assign to read only property 'func' of object '#<ValidatedImmutable>'\n   at http://localhost
    }
  }
  {
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T2783448395",
    "count": 18,
    "time": 199670.00000178814,
    "type": "TEST",
    "status": "FAILED",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the objects",
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should freeze the objects"
    ],
    "error": {
      "message": "Cannot assign to read only property 'grandParent' of object '#<ValidatedImmutable>'",
      "stack": "TypeError: Cannot assign to read only property 'grandParent' of object '#<ValidatedImmutable>'\n   at http://lo
    }
  }
  {
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T1134632789",
    "count": 19,
    "time": 201644.9999064207,
    "type": "TEST",
    "status": "FAILED",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the arrays",
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should freeze the arrays"
    ],
    "error": {
      "message": "Cannot assign to read only property 'arr' of object '#<ValidatedImmutable>'",
      "stack": "TypeError: Cannot assign to read only property 'arr' of object '#<ValidatedImmutable>'\n   at http://localhost:
    }
  }
  {
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T2289548824",

```
    "count": 20,
    "time": 203729.99995015562,
    "type": "TEST",
    "status": "FAILED",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should freeze the values inside the ar
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should freeze the values inside the arrays"
    ],
    "error": {
      "message": "Cannot assign to read only property 'one' of object '#<Immutable>'",
      "stack": "TypeError: Cannot assign to read only property 'one' of object '#<Immutable>'\n    at http://localhost:42001/:66
    }
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T3598350111",
    "count": 21,
    "time": 205804.9999177456,
    "type": "TEST",
    "status": "FAILED",
    "behavior": "given `immutable`, when an immutable is constructed with valid input, it should not allow property deletion",
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with valid input",
      "it should not allow property deletion"
    ],
    "error": {
      "message": "Cannot delete property 'requiredString' of #<ValidatedImmutable>",
      "stack": "TypeError: Cannot delete property 'requiredString' of #<ValidatedImmutable>\n    at http://localhost:42001/:692:
    }
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T2354375029",
    "time": 208329.99981939793,
    "type": "START_TEST",
    "behavior": "given `immutable`, when an immutable is constructed with an instance of another immutable, it should return the
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with an instance of another immutable",
      "it should return the value"
    ]
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T2354375029",
    "count": 22,
    "time": 210489.99996855855,
    "type": "TEST",
    "status": "PASSED",
    "behavior": "given `immutable`, when an immutable is constructed with an instance of another immutable, it should return the
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with an instance of another immutable",
      "it should return the value"
    ],
    "duration": {
      "given": {
        "seconds": 0.00000500003807246685,
        "milliseconds": 0.00500003807246685,
        "microseconds": 5.00003807246685,
        "nanoseconds": 5000.03807246685
      },
      "when": {
        "seconds": 0.00023999996483325958,
        "milliseconds": 0.23999996483325958,
        "microseconds": 239.99996483325958,
        "nanoseconds": 239999.96483325958
      },
      "then": {
        "seconds": 0.0001750001683831215,
        "milliseconds": 0.1750001683831215,
        "microseconds": 175.0001683831215,
        "nanoseconds": 175000.1683831215
      },
      "total": {
        "seconds": 0.0004200001712888479,
        "milliseconds": 0.4200001712888479,
        "microseconds": 420.0001712888479,
        "nanoseconds": 420000.1712888479
      }
    }
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T2789853614",
    "time": 213169.99988630414,
```

```json
    "type": "START_TEST",
    "behavior": "given `immutable`, when an immutable is constructed with invalid input, it should throw",
    "behaviors": [
      "given `immutable`",
      "when an immutable is constructed with invalid input",
      "it should throw"
    ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2789853614",
  "count": 23,
  "time": 215564.99996222556,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an immutable is constructed with invalid input, it should throw",
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with invalid input",
    "it should throw"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": -1,
      "milliseconds": -1,
      "microseconds": -1,
      "nanoseconds": -1
    },
    "then": {
      "seconds": 0.00013500009663403034,
      "milliseconds": 0.13500009663403034,
      "microseconds": 135.00009663403034,
      "nanoseconds": 135000.09663403034
    },
    "total": {
      "seconds": 0.00013500009663403034,
      "milliseconds": 0.13500009663403034,
      "microseconds": 135.00009663403034,
      "nanoseconds": 135000.09663403034
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2676344648",
  "time": 218854.99986819923,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an immutable is constructed with null input, it should throw",
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with null input",
    "it should throw"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2676344648",
  "count": 24,
  "time": 221439.99999389052,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an immutable is constructed with null input, it should throw",
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with null input",
    "it should throw"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": -1,
      "milliseconds": -1,
      "microseconds": -1,
      "nanoseconds": -1
    },
    "then": {
      "seconds": 0.00012500002048909664,
      "milliseconds": 0.12500002048909664,
      "microseconds": 125.00002048909664,
```

      "nanoseconds": 125000.02048909664
    },
    "total": {
      "seconds": 0.00012500002048909664,
      "milliseconds": 0.12500002048909664,
      "microseconds": 125.00002048909664,
      "nanoseconds": 125000.02048909664
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T775081587",
  "time": 224304.99992333353,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an immutable is constructed with undefined input, it should throw",
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with undefined input",
    "it should throw"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T775081587",
  "count": 25,
  "time": 227164.99981470406,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an immutable is constructed with undefined input, it should throw",
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with undefined input",
    "it should throw"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": -1,
      "milliseconds": -1,
      "microseconds": -1,
      "nanoseconds": -1
    },
    "then": {
      "seconds": 0.00011999998241662979,
      "milliseconds": 0.11999998241662979,
      "microseconds": 119.99998241662979,
      "nanoseconds": 119999.98241662979
    },
    "total": {
      "seconds": 0.00011999998241662979,
      "milliseconds": 0.11999998241662979,
      "microseconds": 119.99998241662979,
      "nanoseconds": 119999.98241662979
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T930823784",
  "time": 229904.9999564886,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an immutable is constructed with a valid null value in the input, it should return the
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with a valid null value in the input",
    "it should return the value"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T930823784",
  "count": 26,
  "time": 232379.99994307756,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an immutable is constructed with a valid null value in the input, it should return the
  "behaviors": [
    "given `immutable`",
    "when an immutable is constructed with a valid null value in the input",
    "it should return the value"
  ],
  "duration": {
    "given": {

```json
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": 0.00018500001169741154,
      "milliseconds": 0.18500001169741154,
      "microseconds": 185.00001169741154,
      "nanoseconds": 185000.01169741154
    },
    "then": {
      "seconds": 0.0001650000922381878,
      "milliseconds": 0.1650000922381878,
      "microseconds": 165.0000922381878,
      "nanoseconds": 165000.0922381878
    },
    "total": {
      "seconds": 0.0003500001039355993,
      "milliseconds": 0.3500001039355993,
      "microseconds": 350.0001039355993,
      "nanoseconds": 350000.1039355993
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3908645753",
  "time": 235265.00002481043,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an instance of an immutable is enumerated with Object.keys, prototype functions should
  "behaviors": [
    "given `immutable`",
    "when an instance of an immutable is enumerated with Object.keys",
    "prototype functions should not show up"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3908645753",
  "count": 27,
  "time": 237819.99992206693,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an instance of an immutable is enumerated with Object.keys, prototype functions should
  "behaviors": [
    "given `immutable`",
    "when an instance of an immutable is enumerated with Object.keys",
    "prototype functions should not show up"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": 0.0001799999736249447,
      "milliseconds": 0.1799999736249447,
      "microseconds": 179.9999736249447,
      "nanoseconds": 179999.9736249447
    },
    "then": {
      "seconds": 0.00019999989308416843,
      "milliseconds": 0.19999989308416843,
      "microseconds": 199.99989308416843,
      "nanoseconds": 199999.89308416843
    },
    "total": {
      "seconds": 0.0003799998667091131,
      "milliseconds": 0.3799998667091131,
      "microseconds": 379.9998667091131,
      "nanoseconds": 379999.8667091131
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2542537529",
  "time": 244084.99989658594,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an instance of an immutable is `patch`ed, it should only patch the values that are give
  "behaviors": [
    "given `immutable`",
    "when an instance of an immutable is `patch`ed",
    "it should only patch the values that are given"
  ]
}
{
```

```
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T176323384",
    "time": 246964.99994024634,
    "type": "START_TEST",
    "behavior": "given `immutable`, when an instance of an immutable is `patch`ed, it should return a class that supports `insta
    "behaviors": [
      "given `immutable`",
      "when an instance of an immutable is `patch`ed",
      "it should return a class that supports `instanceof`"
    ]
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T4148801186",
    "time": 249469.99992243946,
    "type": "START_TEST",
    "behavior": "given `immutable`, when an instance of an immutable is `patch`ed, it should not mutate the original object",
    "behaviors": [
      "given `immutable`",
      "when an instance of an immutable is `patch`ed",
      "it should not mutate the original object"
    ]
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T2542537529",
    "count": 28,
    "time": 252604.99981231987,
    "type": "TEST",
    "status": "PASSED",
    "behavior": "given `immutable`, when an instance of an immutable is `patch`ed, it should only patch the values that are give
    "behaviors": [
      "given `immutable`",
      "when an instance of an immutable is `patch`ed",
      "it should only patch the values that are given"
    ],
    "duration": {
      "given": {
        "seconds": 0,
        "milliseconds": 0,
        "microseconds": 0,
        "nanoseconds": 0
      },
      "when": {
        "seconds": 0.0036000001709908247,
        "milliseconds": 3.6000001709908247,
        "microseconds": 3600.0001709908247,
        "nanoseconds": 3600000.1709908247
      },
      "then": {
        "seconds": 0.000709999818354845,
        "milliseconds": 0.709999818354845,
        "microseconds": 709.999818354845,
        "nanoseconds": 709999.818354845
      },
      "total": {
        "seconds": 0.00430999998934567,
        "milliseconds": 4.30999998934567,
        "microseconds": 4309.99998934567,
        "nanoseconds": 4309999.98934567
      }
    }
}
{
    "suiteId": "SDA2A75",
    "batchId": "B1970572985",
    "testId": "T176323384",
    "count": 29,
    "time": 255290.00000096858,
    "type": "TEST",
    "status": "PASSED",
    "behavior": "given `immutable`, when an instance of an immutable is `patch`ed, it should return a class that supports `insta
    "behaviors": [
      "given `immutable`",
      "when an instance of an immutable is `patch`ed",
      "it should return a class that supports `instanceof`"
    ],
    "duration": {
      "given": {
        "seconds": 0,
        "milliseconds": 0,
        "microseconds": 0,
        "nanoseconds": 0
      },
      "when": {
        "seconds": 0.0036000001709908247,
        "milliseconds": 3.6000001709908247,
        "microseconds": 3600.0001709908247,
        "nanoseconds": 3600000.1709908247
      },
```

```json
    "then": {
      "seconds": 0.0033950000070035458,
      "milliseconds": 3.3950000070035458,
      "microseconds": 3395.0000070035458,
      "nanoseconds": 3395000.0070035458
    },
    "total": {
      "seconds": 0.0069950001779943705,
      "milliseconds": 6.9950001779943705,
      "microseconds": 6995.0001779943705,
      "nanoseconds": 6995000.17799437
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T4148801186",
  "count": 30,
  "time": 258224.99999776483,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an instance of an immutable is `patch`ed, it should not mutate the original object",
  "behaviors": [
    "given `immutable`",
    "when an instance of an immutable is `patch`ed",
    "it should not mutate the original object"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": 0.0036000001709908247,
      "milliseconds": 3.6000001709908247,
      "microseconds": 3600.0001709908247,
      "nanoseconds": 3600000.1709908247
    },
    "then": {
      "seconds": 0.006330000003799796,
      "milliseconds": 6.330000003799796,
      "microseconds": 6330.000003799796,
      "nanoseconds": 6330000.003799796
    },
    "total": {
      "seconds": 0.00993000017479062,
      "milliseconds": 9.93000017479062,
      "microseconds": 9930.00017479062,
      "nanoseconds": 9930000.17479062
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T331286067",
  "time": 262094.99989636242,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an instance of an immutable is cast `toObject`, it should return all of the values",
  "behaviors": [
    "given `immutable`",
    "when an instance of an immutable is cast `toObject`",
    "it should return all of the values"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T1131854492",
  "time": 265164.9999897927,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an instance of an immutable is cast `toObject`, the values should be writable",
  "behaviors": [
    "given `immutable`",
    "when an instance of an immutable is cast `toObject`",
    "the values should be writable"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T647668896",
  "time": 268049.99983869493,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an instance of an immutable is cast `toObject`, it should not throw on null values",
  "behaviors": [
    "given `immutable`",
    "when an instance of an immutable is cast `toObject`",
    "it should not throw on null values"
  ]
```

```
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T331286067",
  "count": 31,
  "time": 272804.99995686114,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an instance of an immutable is cast `toObject`, it should return all of the values",
  "behaviors": [
    "given `immutable`",
    "when an instance of an immutable is cast `toObject`",
    "it should return all of the values"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": 0.0006949999369680882,
      "milliseconds": 0.6949999369680882,
      "microseconds": 694.9999369680882,
      "nanoseconds": 694999.9369680882
    },
    "then": {
      "seconds": 0.0018550001550465822,
      "milliseconds": 1.8550001550465822,
      "microseconds": 1855.0001550465822,
      "nanoseconds": 1855000.1550465822
    },
    "total": {
      "seconds": 0.0025500000920146704,
      "milliseconds": 2.5500000920146704,
      "microseconds": 2550.0000920146704,
      "nanoseconds": 2550000.0920146704
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T1131854492",
  "count": 32,
  "time": 276034.9998716265,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an instance of an immutable is cast `toObject`, the values should be writable",
  "behaviors": [
    "given `immutable`",
    "when an instance of an immutable is cast `toObject`",
    "the values should be writable"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": 0.0006949999369680882,
      "milliseconds": 0.6949999369680882,
      "microseconds": 694.9999369680882,
      "nanoseconds": 694999.9369680882
    },
    "then": {
      "seconds": 0.00508500006981194,
      "milliseconds": 5.08500006981194,
      "microseconds": 5085.00006981194,
      "nanoseconds": 5085000.06981194
    },
    "total": {
      "seconds": 0.005780000006780028,
      "milliseconds": 5.780000006780028,
      "microseconds": 5780.000006780028,
      "nanoseconds": 5780000.006780028
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T647668896",
  "count": 33,
  "time": 279429.99987863004,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an instance of an immutable is cast `toObject`, it should not throw on null values",
  "behaviors": [
```

        "given `immutable`",
        "when an instance of an immutable is cast `toObject`",
        "it should not throw on null values"
    ],
    "duration": {
      "given": {
        "seconds": 0,
        "milliseconds": 0,
        "microseconds": 0,
        "nanoseconds": 0
      },
      "when": {
        "seconds": 0.0006949999369680882,
        "milliseconds": 0.6949999369680882,
        "microseconds": 694.9999369680882,
        "nanoseconds": 694999.9369680882
      },
      "then": {
        "seconds": 0.008480000076815486,
        "milliseconds": 8.480000076815486,
        "microseconds": 8480.000076815486,
        "nanoseconds": 8480000.076815486
      },
      "total": {
        "seconds": 0.009175000013783574,
        "milliseconds": 9.175000013783574,
        "microseconds": 9175.000013783574,
        "nanoseconds": 9175000.013783574
      }
    }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2355015786",
  "time": 287389.9999540299,
  "type": "START_TEST",
  "behavior": "given `immutable`, when an instance of an immutable is cast `toObject` with options, and `removeFunctions` is t
  "behaviors": [
    "given `immutable`",
    "when an instance of an immutable is cast `toObject` with options",
    "and `removeFunctions` is true, it should return all of the values, except for functions"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2355015786",
  "count": 34,
  "time": 291969.999903813,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when an instance of an immutable is cast `toObject` with options, and `removeFunctions` is t
  "behaviors": [
    "given `immutable`",
    "when an instance of an immutable is cast `toObject` with options",
    "and `removeFunctions` is true, it should return all of the values, except for functions"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": 0.000654999865218997,
      "milliseconds": 0.654999865218997,
      "microseconds": 654.999865218997,
      "nanoseconds": 654999.865218997
    },
    "then": {
      "seconds": 0.0011400000657886267,
      "milliseconds": 1.1400000657886267,
      "microseconds": 1140.0000657886267,
      "nanoseconds": 1140000.0657886267
    },
    "total": {
      "seconds": 0.0017949999310076237,
      "milliseconds": 1.794999310076237,
      "microseconds": 1794.9999310076237,
      "nanoseconds": 1794999.9310076237
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2937703443",
  "time": 295854.999916628,
  "type": "START_TEST",
  "behavior": "given `immutable`, when the scope of properties used to construct an instance of an immutable change, make sure
  "behaviors": [

```json
      "given `immutable`",
      "when the scope of properties used to construct an instance of an immutable change ",
      "make sure the fixture is in a good state to start"
    ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3044950036",
  "time": 298999.99988265336,
  "type": "START_TEST",
  "behavior": "given `immutable`, when the scope of properties used to construct an instance of an immutable change, the origi
  "behaviors": [
    "given `immutable`",
    "when the scope of properties used to construct an instance of an immutable change ",
    "the original state can change"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3937129236",
  "time": 301949.999993667,
  "type": "START_TEST",
  "behavior": "given `immutable`, when the scope of properties used to construct an instance of an immutable change, reference
  "behaviors": [
    "given `immutable`",
    "when the scope of properties used to construct an instance of an immutable change ",
    "references outside of the scope of the immutable are NOT broken - they CAN mutate"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2937703443",
  "count": 35,
  "time": 306875.0000093132,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when the scope of properties used to construct an instance of an immutable change, make sure
  "behaviors": [
    "given `immutable`",
    "when the scope of properties used to construct an instance of an immutable change ",
    "make sure the fixture is in a good state to start"
  ],
  "duration": {
    "given": {
      "seconds": 0.00000500003807246685,
      "milliseconds": 0.00500003807246685,
      "microseconds": 5.00003807246685,
      "nanoseconds": 5000.03807246685
    },
    "when": {
      "seconds": 0.00009999983012676239,
      "milliseconds": 0.09999983012676239,
      "microseconds": 99.99983012676239,
      "nanoseconds": 99999.83012676239
    },
    "then": {
      "seconds": 0.001980000175535679,
      "milliseconds": 1.9800001755356789,
      "microseconds": 1980.0001755356789,
      "nanoseconds": 1980000.1755356789
    },
    "total": {
      "seconds": 0.002085000043734908,
      "milliseconds": 2.085000043734908,
      "microseconds": 2085.000043734908,
      "nanoseconds": 2085000.043734908
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3044950036",
  "count": 36,
  "time": 310804.99989911914,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when the scope of properties used to construct an instance of an immutable change, the origi
  "behaviors": [
    "given `immutable`",
    "when the scope of properties used to construct an instance of an immutable change ",
    "the original state can change"
  ],
  "duration": {
    "given": {
      "seconds": 0.00000500003807246685,
      "milliseconds": 0.00500003807246685,
      "microseconds": 5.00003807246685,
      "nanoseconds": 5000.03807246685
    },
```

```json
      "when": {
        "seconds": 0.00009999983012676239,
        "milliseconds": 0.09999983012676239,
        "microseconds": 99.99983012676239,
        "nanoseconds": 99999.83012676239
      },
      "then": {
        "seconds": 0.005905000027269125,
        "milliseconds": 5.905000027269125,
        "microseconds": 5905.000027269125,
        "nanoseconds": 5905000.027269125
      },
      "total": {
        "seconds": 0.006009999895468354,
        "milliseconds": 6.009999895468354,
        "microseconds": 6009.999895468354,
        "nanoseconds": 6009999.895468354
      }
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T3937129236",
  "count": 37,
  "time": 314349.9998841435,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when the scope of properties used to construct an instance of an immutable change, reference
  "behaviors": [
    "given `immutable`",
    "when the scope of properties used to construct an instance of an immutable change ",
    "references outside of the scope of the immutable are NOT broken - they CAN mutate"
  ],
  "duration": {
    "given": {
      "seconds": 0.00000500003807246685,
      "milliseconds": 0.00500003807246685,
      "microseconds": 5.00003807246685,
      "nanoseconds": 5000.03807246685
    },
    "when": {
      "seconds": 0.00009999983012676239,
      "milliseconds": 0.09999983012676239,
      "microseconds": 99.99983012676239,
      "nanoseconds": 99999.83012676239
    },
    "then": {
      "seconds": 0.009450000012293458,
      "milliseconds": 9.450000012293458,
      "microseconds": 9450.000012293458,
      "nanoseconds": 9450000.012293458
    },
    "total": {
      "seconds": 0.009554999880492687,
      "milliseconds": 9.554999880492687,
      "microseconds": 9554.999880492687,
      "nanoseconds": 9554999.880492687
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T34662498",
  "time": 318160.0000243634,
  "type": "START_TEST",
  "behavior": "given `immutable`, when a new `Immutable` is constructed/configured, it should let you set the Validator",
  "behaviors": [
    "given `immutable`",
    "when a new `Immutable` is constructed/configured",
    "it should let you set the Validator"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T1772791911",
  "time": 321274.9999947846,
  "type": "START_TEST",
  "behavior": "given `immutable`, when a new `Immutable` is constructed/configured, it should support using JSON Schema",
  "behaviors": [
    "given `immutable`",
    "when a new `Immutable` is constructed/configured",
    "it should support using JSON Schema"
  ]
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T34662498",
  "count": 38,
  "time": 356804.9999885261,
```

```json
    "type": "TEST",
    "status": "PASSED",
    "behavior": "given `immutable`, when a new `Immutable` is constructed/configured, it should let you set the Validator",
    "behaviors": [
      "given `immutable`",
      "when a new `Immutable` is constructed/configured",
      "it should let you set the Validator"
    ],
    "duration": {
      "given": {
        "seconds": 0,
        "milliseconds": 0,
        "microseconds": 0,
        "nanoseconds": 0
      },
      "when": {
        "seconds": -1,
        "milliseconds": -1,
        "microseconds": -1,
        "nanoseconds": -1
      },
      "then": {
        "seconds": 0.03226500004529953,
        "milliseconds": 32.26500004529953,
        "microseconds": 32265.00004529953,
        "nanoseconds": 32265000.04529953
      },
      "total": {
        "seconds": 0.03226500004529953,
        "milliseconds": 32.26500004529953,
        "microseconds": 32265.00004529953,
        "nanoseconds": 32265000.04529953
      }
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T1772791911",
  "count": 39,
  "time": 360919.99989002943,
  "type": "TEST",
  "status": "PASSED",
  "behavior": "given `immutable`, when a new `Immutable` is constructed/configured, it should support using JSON Schema",
  "behaviors": [
    "given `immutable`",
    "when a new `Immutable` is constructed/configured",
    "it should support using JSON Schema"
  ],
  "duration": {
    "given": {
      "seconds": 0,
      "milliseconds": 0,
      "microseconds": 0,
      "nanoseconds": 0
    },
    "when": {
      "seconds": -1,
      "milliseconds": -1,
      "microseconds": -1,
      "nanoseconds": -1
    },
    "then": {
      "seconds": 0.03637499990873039,
      "milliseconds": 36.37499990873039,
      "microseconds": 36374.99990873039,
      "nanoseconds": 36374999.90873039
    },
    "total": {
      "seconds": 0.03637499990873039,
      "milliseconds": 36.37499990873039,
      "microseconds": 36374.99990873039,
      "nanoseconds": 36374999.90873039
    }
  }
}
{
  "suiteId": "SDA2A75",
  "batchId": "B1970572985",
  "testId": "T2036520937",
  "count": 40,
  "time": 364839.99993652105,
  "type": "TEST",
  "status": "SKIPPED",
  "behavior": "given `immutable`, benchmark, compared to regular objects (note ~1/2+ the time is blueprint validation)",
  "behaviors": [
    "given `immutable`",
    "// benchmark",
    "compared to regular objects (note ~1/2+ the time is blueprint validation)"
  ]
}
{
  "suiteId": "SDA2A75",
```

  "time": 369014.9998292327,
  "type": "END",
  "plan": {
    "count": 40,
    "completed": 40,
    "order": [
      "T2794269929",
      "T115500802",
      "T1412403848",
      "T3774403453",
      "T3839474028",
      "T3570737992",
      "T3248676982",
      "T3720816098",
      "T3378731568",
      "T1809451320",
      "T1868228119",
      "T3558144078",
      "T3978851634",
      "T2117899464",
      "T4162694622",
      "T2783448395",
      "T1134632789",
      "T2289548824",
      "T2681499153",
      "T3598350111",
      "T830319501",
      "T2354375029",
      "T2789853614",
      "T2676344648",
      "T775081587",
      "T930823784",
      "T3908645753",
      "T2542537529",
      "T176323384",
      "T4148801186",
      "T331286067",
      "T1131854492",
      "T647668896",
      "T2355015786",
      "T2937703443",
      "T3044950036",
      "T3937129236",
      "T34662498",
      "T1772791911",
      "T2036520937"
    ]
  },
  "totals": {
    "total": 40,
    "passed": 33,
    "skipped": 1,
    "failed": 6,
    "broken": 0,
    "startTime": 71994.99988928437,
    "endTime": 368814.9999361485,
    "duration": {
      "seconds": 0.29682000004686415,
      "milliseconds": 296.82000004686415,
      "microseconds": 296820.00004686415,
      "nanoseconds": 296820000.04686415
    }
  }
}