



0.5 1. Azuki Historical transactions

```
[8]: # Querying the API for transaction data
azuki_tx_url = url + chain_id + "/address" + azuki_address + page_option +
    ↪ api_option
azuki_tx = requests.get(azuki_tx_url).json()

# Convert transactions data to dataframe
azuki_tx_df = pd.DataFrame(azuki_tx['data']['items'], columns =
    ↪ ['to_address_label', 'fees_paid', 'value_quote', 'block_signed_at']).
    ↪ set_index('block_signed_at').sort_index()

azuki_tx_df.head()
```

```
[8]:
```

	to_address_label	fees_paid	value_quote
block_signed_at			
2022-05-02T18:43:51Z	LooksRare: Exchange	22370093235357597	0.000000
2022-05-02T18:47:06Z	None	2448400799338417	0.000000
2022-05-02T18:49:48Z	None	5009529942416780	0.000000
2022-05-02T18:54:24Z	None	5166370935350339	0.000000
2022-05-02T19:06:28Z	None	18519456263386155	86919.574609

0.6 1.a Azuki Historical Sales

```
[9]: # Filter Through data for non null transactions
azuki_sales_df = azuki_tx_df[azuki_tx_df['value_quote'] != 0]
azuki_sales = azuki_sales_df[azuki_sales_df['to_address_label'].notnull()]

# Creating the plot using plotly express
azuki_fig = px.bar(azuki_sales,
    x='to_address_label',
    y='value_quote',
    color='value_quote',
    height=1020,
    width = 1000,
    barmode='overlay',
    labels={'value_quote': 'Amount in USD', 'to_address_label':
    ↪ 'Exchange'},
```